

MONTE CARLO SIMULATIONS OF INTER- AND
INTRA-GRAIN SPIN STRUCTURE OF ISING AND
HEISENBERG MODELS

MARTIN LeBLANC

**MONTE CARLO SIMULATIONS OF INTER- AND INTRA-GRAIN
SPIN STRUCTURE OF ISING AND HEISENBERG MODELS.**

by:

Martin LeBlanc

B. Sc. (spécialisation en physique), Université de Moncton, 2004

**Thesis submitted to the School of Graduate Studies in partial fulfillment of
the requirements of Master of Science.**

Department of Physics and Physical Oceanography
Memorial University of Newfoundland

September 20, 2010

Acknowledgements

I would first like to thank my advisors Dr. Martin Plumer and Dr. John Whitehead for their patience and tremendous help. Their guidance has been instrumental to the accomplishment of this work. Furthermore, Jason Mercer of our research group has also helped me a lot with computer and software related issues and ideas. The examiners of this thesis are likewise to be thanked for their diligent reading and critical comments.

A special thanks is extended to the Department of Physics and Physical Oceanography and its people at Memorial University of Newfoundland. I am also grateful for the financial support received that made this project possible.

Finally, I would like to thank my family for supporting and encouraging me while this work was being done. Similarly, I would like to express my gratitude to my friends for their inspiration throughout this process.

Contents

1	Introduction	1
1.1	Micromagnetics and Magnetic Recording Media	1
1.2	Outline of the Thesis	3
1.3	Magnetic Models and Phase Transitions	5
1.3.1	Ising model	5
1.3.2	Heisenberg model	6
1.3.3	Phase Transitions	7
1.4	Monte Carlo Simulations	9
1.4.1	Monte Carlo	9
1.4.2	Markov Chain Monte Carlo	11
1.4.3	Ergodicity	11
1.4.4	Detailed Balance	11
1.4.5	Metropolis Algorithm	12
1.4.6	Wolff Algorithm	17
1.5	Monte Carlo Simulations of the Ising Model	21
1.6	Monte Carlo Step	23
1.7	Thermodynamic quantities	23
2	Ising Model of Grains	27
2.1	Granular Ising Model	27

2.2	Simulation of Granular Media	30
2.2.1	Initial simulations	30
2.2.2	2D Single-Layer Ising Model Results	35
2.2.3	Results for Multilayer Systems	42
3	Ising Model With External Magnetic Field	50
3.1	M-H Loops: Background	50
3.2	Simulations	52
3.2.1	Homogeneous Ising model	52
3.2.2	Metropolis and Wolff Algorithms	52
3.2.3	Wolff Grain Algorithm	55
3.2.4	Granular Model Simulation results	57
4	Heisenberg Model	60
4.1	Heisenberg Model Simulation	60
4.1.1	Energy	60
4.1.2	Magnetization	61
4.1.3	Specific Heat and Susceptibility	61
4.2	Monte Carlo Algorithms for the Heisenberg Model	62
4.2.1	Metropolis	62
4.2.2	Wolff Algorithm	63
4.3	Heisenberg Model Simulation Results	66
4.3.1	3D Heisenberg Model	67
4.3.2	Thin Film Heisenberg Model	74
5	Discussions and Conclusions	79
5.1	Ising Model	79
5.2	Heisenberg Model	81
5.3	Future Work	82

Bibliography	84
A Code Used in Simulations	87

List of Figures

1.1	Monte Carlo simulation used to determine π with $T = 1000$ and $C = 768$.	10
1.2	Onsager exact solution for the magnetization vs. temperature of the 2D Ising model.	22
2.1	Schematic of the 2D model with 3×3 grains each having 3×3 spins. The labels of J and J' on the figure show locations where these nearest neighbor interactions apply.	28
2.2	Magnetization (a), susceptibility (b) and specific heat (c) vs. temperature for a homogeneous (having no grains) 2D Ising model with the Metropolis algorithm with different values of the lattice size (shown in the legend). Here, $MCS_T = 50000$	31
2.3	Magnetization vs. temperature of the granular Ising model using only the Metropolis algorithm for $J = 0$ and $J = 0.1$. Large fluctuations are seen near the critical temperature. Here, $MCS_T = 50000$	32
2.4	Representation of a single grain showing the effort needed to flip it when using only the Metropolis algorithm. After a single spin is flipped on the right side of the grain, the system might be more inclined to flip it again rather than flip the other spins in the grain.	34
2.5	Total magnetization vs. temperature in the 2D case with $L = 12$ and $L' = 10$ as a function of inter-grain coupling J (values shown in the legend).	35

2.6	Grain magnetization vs. temperature in the 2D case with $L = 12$ for two grain sizes $L' = 5$ and $L' = 10$ and two values of $J = 0.01$ and $J = 0.1$ as shown in the legend.	37
2.7	Susceptibility vs. temperature in the 2D case with $L = 12$ and $L' = 10$ as a function of inter-grain coupling J (values shown in the llegend). . . .	38
2.8	Heat capacity vs. temperature in the 2D case with $L = 12$ and $L' = 10$ as a function of inter-grain coupling J (values shown in the legend). . . .	39
2.9	Spin configuration in the 2D case with $L = 12$ and $L' = 10$ using $J = 0.03$ at $T = 1.79$ for $T_c' \approx 2.0$ and $T_c \approx 0.6$. Up and down spins are shown as black and white points and grain boundaries as grey lines.	40
2.10	T_c vs. J in the 2D case with $L = 12$ and $L' = 5$ for Fig. (a) and $L' = 10$ for Fig. (b).	41
2.11	Magnetization vs. temperature for different values of J shown in the legend. Here, $L' = 10$ and $z = 10$	42
2.12	Susceptibility vs. temperature for different values of J shown in the legend. Here, $L' = 10$ and $z = 10$	43
2.13	Heat capacity vs. temperature for different values of J shown in the legend. Here, $L' = 10$ and $z = 10$. The boxed area approximate the location of the peaks.	44
2.14	Two uniform Ising grains, each having $5 \times 5 \times 5$ spins with one having spins pointing up and the other having spins pointing down. The light red area on the left grain shows the spin that interact with the grain on the right. The area encompasses $L'z = 25$ spins such that here, $J_{\text{eff}} = 25J$	45
2.15	T_c vs. J_{eff} in the multi-layer case for different J , L' and z (values shown in the legend). Straight line has a slope of $2.269 \approx T_c/J_{\text{eff}}$ for the 2D model. . . .	46
2.16	Plot of Eq. 2.8 showing the expected T_c as a function of z . The equation is only valid for $z > 3$	47

3.1	Hysteresis loop of the homogeneous 2D Ising model, showing the coercivity on both sides, where the magnetization sharply changes direction. . . .	51
3.2	M-H loops for the homogeneous 2D Ising model using the Metropolis algorithm with M plotted as a function of H/J' for similar values of T/T_c	53
3.3	H_c/J vs. T/T_c of the homogeneous 2D Ising model calculated with the Metropolis algorithm for two values of $J = J'$ shown in the legend, giving the same profile.	54
3.4	Fig. (a) shows H_c vs. T with only the Metropolis algorithm for the granular multilayer model, falling to zero at T'_c . Fig. (b) shows the first part of an M-H loop with the original Wolff/Metropolis algorithm for $T = 0.5$ having the same parameters as Fig. (a) with $MCS_T = 5000$. Here, $J' = 1$, $J = 0.01$, $L = 8$, $L' = 8$, $z = 8$ and MCS_T is shown in the legend for Fig. (a).	55
3.5	M-H loops calculated using the modified Wolff algorithm with the magnetization plotted as a function of $H_{\text{eff}}/J_{\text{eff}}$ for values of $T/T_c \approx 2/3$ using a range of J_{eff} and L' shown in the legend for the single-layer granular Ising model ($z = 1$). The data for the homogeneous 2D Ising model are also shown.	58
3.6	Plots of the scaled coercive field H_c^c/J_{eff} plotted as a function of T/T_c for several values of J_{eff} and L' shown in the legend for the single-layer granular Ising model ($z = 1$). The data for the homogeneous 2D Ising model are also shown.	59
4.1	Magnetization vs. temperature for different values of $J = J'$ shown in the legend for the homogeneous 3D Heisenberg model with $K = 0$	67

4.2	Granular 80^3 spin Heisenberg cube showing the magnetization with the positive z pointing perpendicular to the top face of the cube. Here, $L' = 5$ and $L = 16$. A purple color represents a negative z spin direction while a blue-green color is the positive z spin direction.	69
4.3	Total magnetization (M) and grain magnetization (M_g) vs. temperature for the 3D granular Heisenberg model. Here, $L = 2$, $L' = 5$, $MCS_T = 25000$, $J = 0.01$ and $K = 10$	70
4.4	Total magnetization vs. temperature for different values of the anisotropy shown in the legend. Here, $L' = 5$, $L = 2$ and $J = 0.01$ for the 3D model.	70
4.5	Magnetization vs. MCS for Fig. (a) $T = 1.5$ and Fig. (b) $T = 1.0$. Here, $L' = 5$, $L = 8$, $J' = 0.01$ and $K = 10$ for the 3D model. The different data series represent independent runs.	71
4.6	Magnetization vs. MCS at $T = 0.6$ and $T = 0.7$ with $K = 0.1$. Here, $L' = 5$, $L = 8$ and $J = 0.01$ for the 3D model.	72
4.7	Magnetization vs. MCS for different values of the number of grains L for $K = 0.01$. Here, $L' = 5$, $T = 0.4$ and $J = 0.01$ for the 3D model. L is shown in the legend.	73
4.8	Magnetization vs. MCS for different values of the temperature with $K = 0.001$. Here, $L' = 5$, $L = 8$ and $J = 0.01$ for the 3D model. T is shown in the legend.	73
4.9	Fig. (a) shows the susceptibility while Fig. (b) shows the specific heat of the granular 3D Heisenberg model for values of K shown in the legend. Here, $L = 4$, $L' = 5$, $MCS_T = 25000$ and $J' = 0.01$	75
4.10	T_c/J vs. K/J of the 2D homogeneous Heisenberg model for $J' = J = 1$ with $L = 80$ and $J' = J = 0.5$ with $L = 160$	76
4.11	T_c vs. J of the 2D homogeneous Heisenberg model for different values of K (shown in the legend). Here, $L = 80$	76

4.12 T_c vs. J_{eff} with different values of K_{eff} shown in the legend for the multi-layer granular Heisenberg model.	77
4.13 T_c vs. J_{eff} with two different values of K/K_{eff} shown in the legend comparing the multilayer granular and 2D homogeneous Heisenberg models.	78
A.1 Input file generator for the Heisenberg model studied with the Monte Carlo method.	87

List of Tables

- 1.1 Possible values of nearest neighbor spins with the resultant energy change.
The second and third columns represent the number of nearest neighbors
of the spin that have spin value +1 or -1. 17
- 2.1 Intra-grain order temperature T'_c at which the peak in the heat capacity
occurs as a function of the grain dimensions L' and z . The values in
parentheses are calculated from the scaling relations from Eqs. 2.8 and 2.10. 48

Abstract

In order to keep supplying computer hard disk drives with more and more storage space, it is essential to have smaller bits. With smaller bits, superparamagnetism, the spontaneous flipping of the magnetic moments in a bit caused by thermal fluctuations, becomes increasingly important and impacts the stability of stored data. Recording media is composed of magnetic grains (usually made of CoCrPt alloys) roughly 10 nm in size from which bits are composed. Most modeling efforts that study magnetic recording media treat the grains as weakly interacting uniformly magnetized objects. In this work, the spin structure internal to a grain is examined along with the impact of varying the relative strengths of intra-grain and inter-grain exchange interactions. The interplay between these two effects needs to be examined for a greater understanding of superparamagnetism as well as for the applications of the proposed Heat Assisted Magnetic Recording (HAMR) technology where thermal fluctuations facilitate head-field induced bit reversal in high anisotropy media.

Simulations using the Monte Carlo method (with cluster-flipping algorithms) are performed on a 2D single-layer and multilayer Ising model with a strong intra-grain exchange interaction J as well as a weak inter-grain exchange J' . A strong deviation from traditional behavior is found when J'/J is significant. M-H hysteresis loops are also calculated and the coercivity, H_c , is estimated. A large value represents a strong resilience to the superparamagnetic effect. It is found that taking into account the internal degrees of freedom has a significant effect on H_c . As the Ising model serves only as an approximation, preliminary simulations are also reported on a more realistic Heisenberg model with uniaxial anisotropy.

Key Words: Ising model, Heisenberg model, Monte Carlo Simulation

Chapter 1

Introduction

1.1 Micromagnetics and Magnetic Recording Media

Micromagnetics, as the name suggests, is a field which deals with phenomena related to magnetism at sub-micrometer length scales. It typically uses the Landau-Lifshitz-Gilbert (LLG) stochastic differential equation to obtain the magnetization of a system under different situations. At these scales, there are many interactions in the energy E to take into account, namely the exchange energy, the anisotropy energy, the Zeeman energy (interaction with an external magnetic field), the dipolar energy and temperature effects through a Langevin stochastic term.

The LLG equation is named after Lev Landau, Evgeny Lifshitz and T. L. Gilbert and is often given as

$$\frac{\partial \mathbf{m}}{\partial t} = -\gamma \mathbf{m} \times \mathbf{H}_{\text{eff}} + \alpha \mathbf{m} \times \frac{\partial \mathbf{m}}{\partial t}, \quad (1.1)$$

where \mathbf{m} is the magnetization, $\mathbf{H}_{\text{eff}} = -\frac{\delta E}{\delta \mathbf{m}}$ is the effective magnetic field (due to the different interactions) and α and γ are a damping parameter and the electron gyroscopic ratio, respectively. This allows a realistic description of the system, from which the time-dependent magnetization can be calculated as a function of the effective magnetic field,

H_{eff} . This effective field can become quite complicated which makes the calculation of the magnetization non-trivial.

Micromagnetics has had, for the past twenty years, considerable success at modeling magnetic recording media [1, 2] as it allows for a fairly accurate modelling of magnetic grains used in magnetic recording media, such as hard disk drives. A fundamental assumption of micromagnetics is that there are well defined regions with a uniform magnetization having a constant magnitude $|m|$. In the case of current recording media based on highly anisotropic cobalt alloys, these regions are taken to be the magnetically separated grains which are typically 8-9 nm in diameter and are composed of hundreds to thousands of atomic spins. For conventional recording media, the consideration of grains as uniformly magnetized may be justified by the fact that the intra-grain (inside grains) spin-spin ferromagnetic exchange interaction is typically 10-100 times larger than inter-grain (between grains) exchange. In addition, temperatures relevant for typical recording processes ($\sim 320\text{K}$) are about one-quarter of the Curie temperature for cobalt ($T_c = 1400\text{K}$) so that thermal effects are not obviously important.

Due to reductions in bit size, the need for more accurate models, and the increasingly important role of thermal effects there has been an exploration of modified approaches which go beyond the assumption of a uniformly magnetized grain and consider effects due to intra-grain spin degrees of freedom [3, 4, 5, 6]. As grain sizes shrink, surface spins can play an increasingly important role in determining reversal mechanisms. This is due not only to modifications in surface exchange interactions arising from simple geometrical arguments but also to a reduction of surface-spin anisotropy. These effects can lead to modifications in grain magnetic moments and magnetic field-induced reversal mechanisms important for the recording process [7, 8, 9, 10, 11, 12].

It is anticipated that new technologies will be introduced into the recording process within a few years in an effort to ensure a continuation of bit size reduction [13]. One of the more promising is considered to be heat assisted magnetic recording (HAMR) [14] where bit (and grain) reversal in an applied head field is facilitated by thermal

fluctuations. This allows for the possibility of higher anisotropy, smaller grain media thereby avoiding the superparamagnetic effect and spontaneous decay of stored information [15]. The HAMR concept is being explored at temperatures near the Curie point of cobalt where the grain magnetization is reduced to near zero. While thermal effects can be incorporated into the LLG formalism by the introduction of a stochastic term, the thermal fluctuations only affect the direction of the grain magnetization, not its magnitude. One phenomenological approach to include intra-grain degrees of freedom is to assign a temperature dependence for the magnetization and single-ion anisotropy which mimics bulk cobalt [16, 17]. A more sophisticated technique is to incorporate longitudinal grain-moment magnitude dynamics into the LLG equations [18]. Nonlocal reversal mechanisms where grain surface spins initiate the process are not captured in these models. A detailed account of these known effects are expected to play an increasingly important role in technologies and future modelling efforts.

1.2 Outline of the Thesis

In this work, the limitations of assuming uniformly magnetized grains is explored within the context of a simple model that includes explicitly the internal atomic spins of the grains in magnetic recording media. Monte Carlo simulations are performed on a 2D system assuming grains composed of $L' \times L' \times z$ Ising spins with strong nearest-neighbor intra-grain (spin-spin) exchange interactions J' and weaker inter-grain exchange J between the 2D lattice of $L \times L$ grains. The Ising model may be viewed as a simple first approximation to a more realistic Heisenberg model with the strong anisotropy present in cobalt-based alloys. In order to address equilibration issues that arise as a consequence of the large difference in the interaction energies in this model, the simulations utilize a combination of a cluster-flip Wolff algorithm and a Metropolis algorithm. A number of thermodynamic quantities are evaluated that reflect the degree of intra-grain and inter-grain ordering of the spins. From these, regimes where a uniformly magne-

tized grains approximation is valid can be distinguished. Hysteresis associated with the reversal of the magnetization in an applied field for temperature below the critical temperature is studied and the impact of the intra-grain spins in determining the coercivity is examined. Lastly, simulations are performed on the zero-field anisotropic Heisenberg model constructed in a similar way to the Ising case.

In the remainder of this chapter, the Ising and Heisenberg magnetic models which are used in this work are explained along with some background information on phase transitions as they relate to magnetic models. Monte Carlo simulations are explained and it is discussed how they can be used to simulate magnetic models. Finally, the Metropolis and Wolff algorithms that were used to obtain the simulation results are introduced with an emphasis on how to use them in simulations.

Chap. 2 explores the Ising model with grains with an explanation on the terminology used and how the grains are simulated. An explanation is given on how the Metropolis and Wolff algorithms are modified to be used for a granular model. Results are given for the 2D Ising model, leading up to how the critical temperature reacts in an unexpected way as different coupling strengths are studied. This is then redone for a multilayer model to better represent thin films of magnetic material. An equation is derived that can be used to estimate the critical temperature as a function of the simulated grain size and number of layers for the granular Ising model. The work is done with no external magnetic field present.

Chap. 3 also studies the granular Ising model, but with an external magnetic field. A modified Wolff algorithm is explained that was needed to obtain the coercivity H_c for the Ising model. M-H loops were obtained of the 2D Ising model with grains as a way to study the coercivity H_c as a function of temperature and the results are compared to the homogeneous (no grains) model.

Chap. 4 explores the simulation of the anisotropic Heisenberg model. An explanation is given on how the Metropolis and the Wolff algorithms can be applied to the Heisenberg model. Results are given for the granular Heisenberg model and fluctuations seen for

higher anisotropies are studied by looking at the magnetization as a result of simulation time. Results of the critical temperature for the multilayer model as a function of the inter-grain coupling constant and the anisotropy are obtained with comparison to the homogeneous model.

Finally, a discussion on the results and conclusions drawn from them is presented in Chap. 5. Possibilities for future work are also mentioned.

1.3 Magnetic Models and Phase Transitions

1.3.1 Ising model

The simplest and by far the most famous model that can be conceived to study magnetic systems is the spin-1/2 Ising model [19]. In this model, a lattice of discrete spins are represented as each having a magnetic moment of +1 or -1. In its simplest form, it has only nearest neighbor interactions. In one dimension, it consists of a chain of L spins where each spin interacts with its two nearest neighbors. In two dimensions, the square lattice Ising model has $L \times L$ spins in which each spin interacts with its four nearest neighbors, two in each lateral direction, has been studied theoretically using a variety of methods such as low and high temperature series expansions, transfer matrixes, mean field theory and Green's function theory.

The Ising model was first formulated by Ernst Ising in 1925 [20] who presented the exact solution for the 1D case in his PhD thesis. It was not until 1944 that Lars Onsager managed to analytically solve the square lattice 2D Ising model in zero field. So far, no-one has been able to solve the 2D model with external magnetic field or the 3D case for this simplest of magnetic models.

The most common of Ising models is ferromagnetic with a constant exchange energy J and external magnetic field H given by the Hamiltonian

$$\mathcal{H} = -J \sum_{\langle ij \rangle} S_i S_j - H \sum_i S_i. \quad (1.2)$$

where $\langle ij \rangle$ represents nearest-neighbor interactions only. However, taking into account site- or spin-specific exchange parameters, it can be given as

$$\mathcal{H} = - \sum_{ij} J_{ij} S_i S_j - H \sum_i S_i. \quad (1.3)$$

Here, the S_i and the S_j denote the spin variables at different lattice sites that can take the value of $+1$ or -1 . Each interaction (bond) between two nearest neighbor is counted once so that a 2D model with 100 spins has 200 bonds. J_{ij} is the exchange coupling between spins and H is the external magnetic field. The standard ferromagnetic Ising model consists of a nearest-neighbor interaction with $J_{ij} = J > 0$, while a nearest-neighbor interaction $J_{ij} = J < 0$ defines the antiferromagnetic Ising model. The case of $J_{ij} = 0$ is called the non-interacting Ising model, as spins do not interact with their neighbors. The J_{ij} can be changed depending on the model, as, for example, having a random J_{ij} for different bonds is one model for a spin glass. A number of other modifications can be done to this model to represent different systems, such as adding next-nearest-neighbor interactions or adding dipole interactions wherein each spins affects other spins with an interaction that depends on the displacement vector separating the spins and their orientation.

1.3.2 Heisenberg model

The classical Heisenberg model [21] is defined by a Hamiltonian that is similar to that of the Ising model

$$\mathcal{H} = - \sum_{\langle ij \rangle} J_{ij} \mathbf{S}_i \cdot \mathbf{S}_j - \sum_i \mathbf{H} \cdot \mathbf{S}_i - K \sum_i (S_i^z)^2. \quad (1.4)$$

where the spin variables \mathbf{S}_i denote 3D vectors with $|\mathbf{S}_i| = S$. Here, \mathbf{H} is the external magnetic field strength and K is the single-ion magnetocrystalline anisotropy of the system. The anisotropy represents the tendency for the magnetic moments to align in a certain direction. In the case of thin films, this is often the $\hat{\mathbf{z}}$ direction, perpendicular to the film plane. $K > 0$ causes the spins to orientate along $\hat{\mathbf{z}}$ while $K < 0$ causes the spins to orientate in the plane perpendicular to $\hat{\mathbf{z}}$. Another type of anisotropy sometimes used is an exchange anisotropy with terms of the type $J_z \sum_{\langle ij \rangle} S_i^z S_j^z$.

The limit of infinite single-ion anisotropy, $K \rightarrow \infty$, constrains the spins to point in one direction and the Heisenberg model reduces to the Ising model. If, instead, $K \rightarrow -\infty$, the spins are constrained to lie on the plane normal to $\hat{\mathbf{z}}$ and the Heisenberg model reduces to the XY model. As the Ising model allows spins to vary in a single dimension and the Heisenberg model allows the spins to vary in three dimensions, the XY model allows spins to rotate in the 2D XY plane. As such, varying the anisotropy within the Heisenberg model allows the system to approach the behavior of the Ising and XY models.

1.3.3 Phase Transitions

An order parameter is defined as a quantity which is zero in one phase and non-zero in another. The behavior of this quantity as a function of some parameter like the temperature determines the point at which the system undergoes a phase transition and switches between phases.

There are two main classes of phase transitions [22], *first-order phase transitions* and *continuous phase transitions* (also sometimes called second order phase transitions). In a first order phase transition, the order parameter jumps discontinuously from zero to a certain value and the system can have a co-existence of different phases, such as water at its boiling point. In a continuous phase transition, as the name suggests, the order parameter goes from zero to a non-zero value continuously at the critical temperature and there is a divergence in certain properties, such as the susceptibility and the specific

heat. There are many theories to study these phenomena, such as mean field theory or the phenomenological Landau theory of second order phase transitions, which can provide a qualitative description of the transition.

For the magnetic systems under study here, the second order ferromagnetic phase transitions of the Ising and Heisenberg models are considered. The order parameter is the magnetization, which falls off to zero at the critical temperature T_c , called the Curie temperature for ferromagnetic systems. In the case of the Ising and Heisenberg antiferromagnet, the order parameter is the staggered magnetization.

When studying continuous phase transitions, a number of quantities exhibit singularities at the critical point. These singularities are characterized by a set of critical exponents. For example, in a magnetic system some of the exponents defined are for the zero field specific heat, magnetization and susceptibility, known as α , β and γ respectively. These would be defined like

$$C \propto |T - T_c|^{-\alpha}, \quad (1.5)$$

which describe the behavior of the specific heat C near the critical temperature.

These exponents are interesting because of universality: systems that appear different but have a few essential properties in common possess the same critical exponent. A universality class comprises systems which exhibit this behavior, e.g. certain ferromagnetic systems and the liquid-gas phase transition for fluids. The thermodynamic properties of the system would seem to depend only on the few parameters, such as dimensionality and symmetry.

1.4 Monte Carlo Simulations

1.4.1 Monte Carlo

Monte Carlo simulations are used in many diverse areas of science, from the study of fluids, finances, artificial intelligence, differential equations to chemistry or astronomy. The general idea is simply to generate random quantities as input in such a way as to probe the system of study and gather useful information [23]. For example, one way to approximately calculate π with a Monte Carlo simulation is to draw a circle of radius $\frac{L}{2}$ inside a square of length L . The area of the circle being $\pi(\frac{L}{2})^2$ and the area of the square being L^2 , the ratio of these two areas is given as $\frac{\pi L^2/4}{L^2} = \frac{\pi}{4}$. By generating T uniform random locations inside the square and then counting the amount C that fall within the circle, the ratio C/T will approximate $\pi/4$. For example, if in a simulation $T = 1000$ and $C = 768$, $4C/T = 4 * 768/1000 = 3.072$, which approaches the true value of $\pi \approx 3.14$. The representation of this example is shown in Fig. 1.1. To get a closer estimate, the simulation can be repeated multiple times and the result averaged, or a higher value of T can be used.

From this fairly simple example, the basics of using Monte Carlo simulations are shown. Even with a thousand random values as input, the result obtained was only approximately close to π , in this case obtaining a value only accurate to the first digit. To obtain many more accurate significant figures, a very high value of T is needed which would become computationally expensive very quickly. As such, for cases like these the Monte Carlo method can provide a *first stab* approach at a problem using a very simple principle. This is not to say that it is ineffective as, for higher dimensions, Monte Carlo is a very efficient method of calculating integrals.

Monte Carlo in Statistical Mechanics

The Monte Carlo method is widely used in statistical mechanics, where the internal energy of the system can be computed based on the states of the particles in the system.

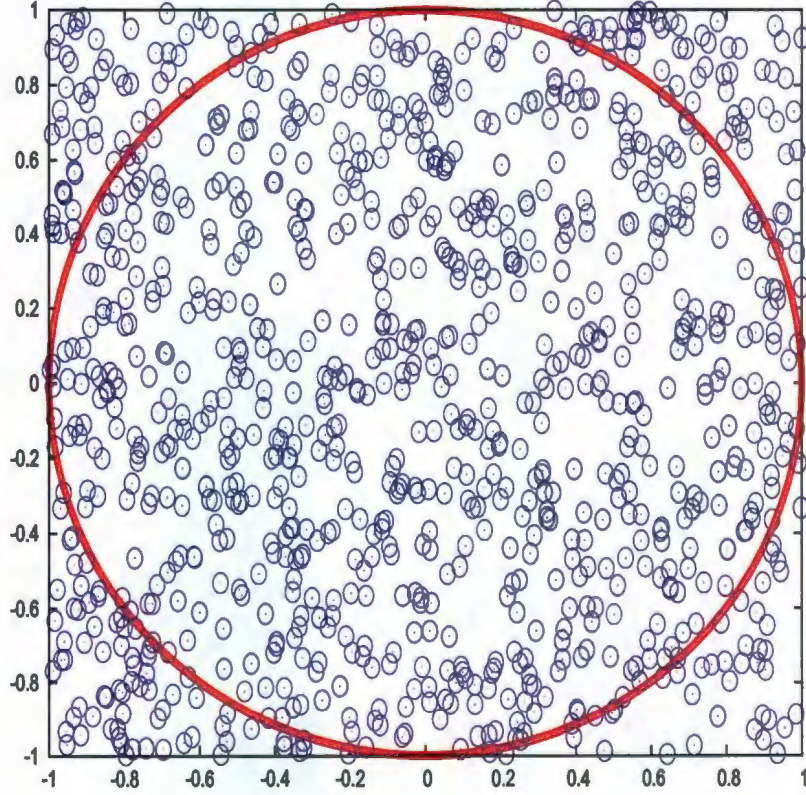


Figure 1.1: Monte Carlo simulation used to determine π with $T = 1000$ and $C = 768$.

With Maxwell-Boltzmann statistics, the probability at equilibrium that the system will be in a specific state a is given by

$$p_a = \frac{1}{Z} e^{-\beta E_a}, \quad (1.6)$$

where $\beta = \frac{1}{kT}$ with k the Boltzmann constant ($1.38 \times 10^{-23} \text{ JK}^{-1}$) and T the temperature, E_a is the internal energy for the state a and Z is the partition function

$$Z = \sum_a e^{-\beta E_a}. \quad (1.7)$$

In the study of magnetic systems, several quantities are of value to study, such as

the magnetization or the energy. The expectation value (or average over the different states of the system) of such quantities Q in equilibrium is given by

$$\langle Q \rangle = \frac{1}{Z} \sum_a Q_a e^{-\beta E_a}. \quad (1.8)$$

The average of Q over different states, $\langle Q \rangle$ is often written simply Q when it is obvious that an average is being taken.

1.4.2 Markov Chain Monte Carlo

When a process follows a sequential ordering and a new state b can be obtained randomly from the current state a without access to all the other previous states, it is called a Markov chain [24]. Markov chain Monte Carlo methods are algorithms for which a specific probability distribution is followed, such as the Boltzmann distribution. The Markov chain states are chosen to represent the system of study and there are a number of other constraints that are imposed so as to be useful.

1.4.3 Ergodicity

A process or algorithm is said to be ergodic, or to satisfy ergodicity, if any state can be obtained from any other state, provided a sufficiently long simulation time to sample all possible states of a system. This property must be satisfied to get a useful algorithm. As such, from any state a , there must be a path, no matter how long it takes, to any other state b in the Markov chain. Conversely, the system must also be able to go back from b to a . This is usually fairly simple to satisfy for a given Monte Carlo algorithm with an appropriate choice of transition probability, but it is crucial to take these factors into consideration.

1.4.4 Detailed Balance

The transition probabilities used for the system must satisfy detailed balance, given by

$$\frac{P(a, b)}{P(b, a)} = \frac{p_b}{p_a}, \quad (1.9)$$

where $P(a, b)$ ($P(b, a)$) is the probability that the system will generate state b (a) from state a (b) and transition to it, and the p_a and p_b are the probabilities of occupation at equilibrium. Working with classical systems, the equilibrium distribution used is the Boltzmann distribution, which is given by Eq. 1.6. Thus, to satisfy detailed balance with a system following the Boltzmann distribution, Eq. 1.9 becomes

$$\frac{P(a, b)}{P(b, a)} = e^{-\beta(\Delta E)}, \quad (1.10)$$

where $\Delta E \equiv E_b - E_a$.

The transition probability $P(a, b)$ has two parts: the selection probability $g(a, b)$ and the acceptance ratio $A(a, b)$, such that $P(a, b) = g(a, b)A(a, b)$. The selection probability is the probability that the algorithm generates a specific state b from state a and the acceptance ratio is the probability for the system to change from state a to this state b . As an example of the selection probability, an algorithm could be constructed that selects a random spin of a 2D square-lattice Ising model with N (even) spins such that spins on the left side of the lattice are selected twice as often as spins on the right side. The selection probability $g(a, b)$ of selecting a spin on the favored side of the lattice is then $g(a, b) = \frac{4}{3N}$ while the selection probability for a spin belonging to the other side is $g(a, b) = \frac{2}{3N}$. The acceptance ratio is then the probability of the algorithm doing something to the spin, such as changing its value. Eq. 1.10 is then used to determine this probability when using a system following the Boltzmann distribution.

1.4.5 Metropolis Algorithm

The condition of detailed balance does not uniquely determine what transition probabilities should be used. As such, this gives some flexibility to develop an efficient algorithm that applies to the problem at hand, so long as the ergodicity and detailed balance

conditions are satisfied. For example, any implementation of $P(a, b)$ and $P(b, a)$ that satisfies Eq. 1.10 will give a correct result. However, most choices are simply not efficient and will not be useful.

The Metropolis-Hastings algorithm [25], named after Nicholas Metropolis and W. Keith Hastings, is a particular choice of the transition probabilities so as to be very efficient. The original algorithm was developed in 1953 by N. Metropolis using the Boltzmann distribution and was extended to the general case in 1970 by W. K. Hastings. It is referred here as the Metropolis algorithm as only the Boltzmann distribution is used. It is the most commonly used Markov chain Monte Carlo method due to its simplicity and efficiency.

The Metropolis algorithm has single-spin-flip dynamics, as it only considers a single spin at a time. This is not a requirement and there exists other algorithms that consider more than one spin, such as the Wolff algorithm, explained in Sec. 1.4.6.

The derivation of the Metropolis algorithm follows from a few simple steps [24]. From state a , there are N possible states that can be reached after one flip to create different states. The probability to create a specific state b from a is thus $g(a, b) = \frac{1}{N}$ as they are all equally favored; the probability of creating state a from b is also the same. The condition of detailed balance, which follows from Sec. 1.4.4, can then be stated as

$$\frac{P(a, b)}{P(b, a)} = \frac{A(a, b)g(a, b)}{A(b, a)g(b, a)} = \frac{A(a, b)}{A(b, a)} = e^{-\beta\Delta E}. \quad (1.11)$$

The choice of acceptance ratio can be made in almost any fashion, so long as this equation is obeyed; however, a low acceptance ratio would lead to many wasted calculations, and as such a large acceptance ratio is therefore generally more efficient. For this purpose, the larger of the two acceptance ratios is usually taken to be unity while the other one is adjusted accordingly. With this single-spin flip algorithm, there are three situations: the energy of the new state can be either smaller or larger than the one of the current state or remain the same. If $\Delta E > 0$, $e^{-\beta\Delta E}$ will be smaller than unity, as

$\beta > 0$. This means that $P(b, a) > P(a, b)$, so $P(b, a)$ is set to 1 and $P(a, b) = e^{-\beta\Delta E}$. If, however, $\Delta E < 0$, the exponential will be greater than unity and a similar logic is applied: $P(a, b)$ is set to 1 and $P(b, a) = e^{-\beta\Delta E}$. Having $\Delta E = 0$ has $P(a, b) = P(b, a) = 1$, such that the spin is flipped with no change in energy.

What this means is that when the spin flip reduces the energy ($\Delta E < 0$), the probability of accepting the new configuration is 1. On the other hand, when the spin flip increases the energy, the new configuration is accepted with a probability given by the weight $w = e^{-\beta\Delta E}$. To implement this, a random number r is chosen such that $0 \leq r < 1$. If $r < w$, the spin is flipped. This is the Metropolis algorithm, which has seen remarkable success. Other implementations of the acceptance ratio can be used if they prove to be more applicable to the problem at hand, but the Metropolis algorithm has shown itself to generally be very efficient. Ergodicity is assured by being able to flip one spin at a time, indefinitely, to go from any state to another.

As a summary to the usage of the Metropolis algorithm in Monte Carlo simulations, the following steps are done for one application of the algorithm:

1. Calculate the energy of the system.
2. Pick a random spin on the lattice.
3. Calculate the energy if the spin is flipped and ΔE , the difference in energy between this energy and the previous one.
4. If $\Delta E \leq 0$, the spin is flipped. Otherwise, the next step is followed.
5. Calculate the Boltzmann weight, $w = e^{-\beta\Delta E}$.
6. Generate a random number $0 \leq r < 1$.
7. If $r < w$, flip the spin. If not, no change is made.

As can be seen, this algorithm involves calculating the Boltzmann weight w every time a spin is considered. This can easily become computationally expensive, as calcu-

lating an exponential is an expensive operation. Luckily, for the Ising model, there is a shortcut that can be used to help speed calculations. The weights can be calculated in advance and placed inside a lookup table. It is then only a matter of figuring out the current spin's situation and looking at the precalculated weights to gather what is the value of w .

A first implementation of the Metropolis algorithm into a computer program will almost surely involve calculating the system's entire energy before flipping the spin, the system's entire energy afterwards and finding ΔE by comparing the two. While this does work, it is another source of inefficiency as it involves many unused calculations. Consider the effect of flipping a single spin S_i to a new spin value S'_i on the total energy of the system. The only thing that changes is S_i which becomes S'_i , but $S'_i = -S_i$ for the Ising model. From Eq. 1.3, the spin's flip only affects its nearest neighbors from the first term (nearest neighbor interaction) and itself from the last term (external magnetic field).

To help with speed calculations, the system's energy and magnetization can be calculated before applying the Metropolis algorithm many times. After each spin flip, the change is applied to these quantities instead of calculating them again. The single spin flip change used is computationally inexpensive compared to calculating the contribution of the entire lattice again, which helps speed up the runs.

The exchange energy term due to the nearest neighbors will have (in the 2D case in this example) four contributions to it, consisting of all the nearest neighbors. From the Hamiltonian, the energy associated with nearest-neighbor interactions is

$$E_J = -J \sum_{\langle ij \rangle} S_i S_j. \quad (1.12)$$

The only terms that are affected are the four for which $\langle ij \rangle$ involves S_i . These can be explicitly given as

$$E_{J_i} = -J \sum_{k=1}^4 S_i S_{i+k} = -JS_i \sum_{k=1}^4 S_{i+k}, \quad (1.13)$$

where the k runs over the four nearest neighbors, the two in the x direction and the two in the y direction. After flipping the spin, the energy E_{J_i} becomes E'_{J_i} , given by

$$E'_{J_i} = -JS'_i \sum_{k=1}^4 S_{i+k} = JS_i \sum_{k=1}^4 S_{i+k}. \quad (1.14)$$

The difference in energy is then given by

$$\Delta E_{J_i} = E'_{J_i} - E_{J_i} = JS_i \sum_{k=1}^4 S_{i+k} - (-JS_i \sum_{k=1}^4 S_{i+k}) = 2JS_i \sum_{k=1}^4 S_{i+k}. \quad (1.15)$$

The Zeeman term due to the external magnetic field will have a contribution to the total energy due to $E_{H_i} = -HS_i$ becoming $E'_{H_i} = -HS'_i = HS_i$. It is then very simple to calculate the change in energy due to the Zeeman term

$$\Delta E_{H_i} = E'_{H_i} - E_{H_i} = HS_i - -HS_i = 2HS_i. \quad (1.16)$$

Finally,

$$\Delta E = \Delta E_{J_i} + \Delta E_{H_i}. \quad (1.17)$$

Since all the spins can only take a value of $+1$ or -1 , one can calculate all the possible ΔE . The spin S_i has two choices and can have neighbors that have the spin values given in Table 1.1.

With a knowledge of the current temperature T and thus of β , it is then very simple to calculate and store these few $w = e^{-\beta \Delta E}$ before applying the Metropolis algorithm many times. This analysis can be extended to consider a different amount of nearest neighbors, such as six for a 3D system.

S_i	+1 neighbors	-1 neighbors	$\sum_{k=1}^4 S_{i+k}$	ΔE_{J_i}	ΔE_{H_i}	ΔE
+1	0	4	-4	$-8J$	$2H$	$-8J + 2H$
+1	1	3	-2	$-4J$	$2H$	$-4J + 2H$
+1	2	2	0	0	$2H$	$2H$
+1	3	1	2	$4J$	$2H$	$4J + 2H$
+1	4	0	4	$8J$	$2H$	$8J + 2H$
-1	0	4	-4	$8J$	$-2H$	$8J - 2H$
-1	1	3	-2	$4J$	$-2H$	$4J - 2H$
-1	2	2	0	0	$-2H$	$-2H$
-1	3	1	2	$-4J$	$-2H$	$-4J - 2H$
-1	4	0	4	$-8J$	$-2H$	$-8J - 2H$

Table 1.1: Possible values of nearest neighbor spins with the resultant energy change. The second and third columns represent the number of nearest neighbors of the spin that have spin value +1 or -1.

1.4.6 Wolff Algorithm

While the Metropolis algorithm is a good general purpose procedure, due to the large thermal fluctuations that arise in the vicinity of T_c , the number of applications of the algorithm required to equilibrate the system increases substantially, an effect referred to as critical slowing down. In the case of finite size systems, the correlation length is limited by the system size, which gives rise to size dependent effects in many of the quantities of interest close to the critical temperature and an increase in the time required to equilibrate the system with increasing size.

It has been shown that the difficulties associated with critical slowing down may be avoided by using an algorithm that allows "moves" that involve large numbers of spins. Instead of flipping a single spin at a time, these so-called cluster algorithms build virtual clusters consisting of a select amount of spins and flip them. While these algorithms address the issue of the long equilibration time near the critical temperature, they are not as efficient when studying the system at temperatures away from it.

A widely used cluster algorithm is the Swendsen-Wang (1987) algorithm, which, in a single step, creates many virtual clusters and flips a proportion of them; it can easily be shown to satisfy ergodicity and detailed balance. Another algorithm developed

two years later by Ulli Wolff [26] was based on the Swendsen-Wang algorithm with some improvements which make it more efficient, especially at high temperatures. The Wolff algorithm is the most commonly used cluster algorithm, as it is very efficient at equilibrating systems.

Above the critical temperature, the spins are mostly random and uncorrelated, while below the critical temperature the spins are mostly correlated and everything points mostly in the same direction. The Wolff algorithm takes advantage of this and builds a virtual cluster which depends on the current temperature. At high temperatures, the clusters will be small while for low temperatures the cluster will be very big, often times expanding to encompass the entire lattice. Picking a random seed spin, the cluster is made up of spins that all point in the same direction. To account for the change in correlation link by the temperature, all similarly pointing spins are not added, but have a probability of being added which depends on the temperature.

The derivation of the Wolff algorithm for the Ising model [24] also follows from the discussion on detailed balance from Sec. 1.4.4, but is a bit more complicated since the algorithm does not use single-spin dynamics. As a consequence, the selection probabilities $g(a, b)$ are not simply going to be $\frac{1}{N}$. For a specific cluster, there are only two moves to consider: going from state a to b by selecting a random seed spin, building the cluster and flipping it and the reverse move involving going from state b to a where the same seed spin is used and the same cluster is built. The only difference between these two moves is at the cluster border and the interaction between those spins. As such, even though a cluster can be built in multiple ways (since a spin can be considered more than once), especially for 2D and 3D systems, the selection probabilities $g(a, b)$ and $g(b, a)$ will always share a common factor. While these probabilities can become quite complicated for large systems and big clusters, as the seed spin is chosen at random, $\frac{g(a, b)}{g(b, a)}$ depends only on the boundary spins and everything else cancels out.

Spins are added to the cluster with a certain probability P_J , which means that rejection has a probability $1 - P_J$. Naming m the number of bonds that are in state

a between spins at the boundary of the cluster and same-valued spins just outside the clusters and n the number of such bonds that are in state b , the probability of not adding all m spins to the cluster in state a is $(1 - P_J)^m$ and the probability of not adding all n spins to the cluster in state b is $(1 - P_J)^n$. The probability $g(a, b)$ of creating state b from state a is thus $g_0(1 - P_J)^m$ and $g(b, a)$ for the reverse move is $g_0(1 - P_J)^n$, where g_0 is the same for both, a factor that depends on the size of the cluster and the size of the lattice. Detailed balance can then be stated as

$$\frac{P(a, b)}{P(b, a)} = \frac{g_0(1 - P_J)^m A(a, b)}{g_0(1 - P_J)^n A(b, a)} = (1 - P_J)^{m-n} \frac{A(a, b)}{A(b, a)} = e^{-\beta \Delta E}. \quad (1.18)$$

To figure out the ΔE , it is simply noted that upon flipping the cluster, only the boundary spins' interactions will affect the nearest neighbor exchange energy. Furthermore, the effect on the magnetic field energy can be easily calculated based on how many spins are flipped. Setting k as the cluster size, this last one becomes

$$\Delta E_H = -kHS'_i - -kHS_i = kHS_i + kHS_i = 2kHS_i, \quad (1.19)$$

where S_i is the spin value inside the cluster before flipping and S'_i the value after flipping. For ΔE_J , there is the contribution from the m bonds and the n bonds. The total number of spins touching the cluster boundary is $m + n$, with m bonds having the same spin direction as S_i and n bonds having $S'_i = -S_i$. These m bonds each contribute an energy of $-J(S_i)^2 = -J$ while the n bonds contribute J . Upon flipping the cluster, the energy contributions will switch and so:

$$\Delta E_J = E'_J - E_J = (m(J) + n(-J)) - (m(-J) + n(J)) = 2J(m - n). \quad (1.20)$$

Eq. 1.18 then becomes

$$\frac{A(a,b)}{A(b,a)} = (1-P_J)^{-(m-n)} e^{-\beta(2J(m-n))} e^{-\beta 2kHS_i} = (e^{2\beta J}(1-P_J))^{-(m-n)} e^{-2\beta kHS_i}. \quad (1.21)$$

One choice of P_J that greatly simplifies this can be immediately seen: $P_J = 1 - e^{-2\beta J}$. This completely removes the dependence on m and n and reduces the problem to

$$\frac{A(a,b)}{A(b,a)} = e^{-2\beta kHS_i}. \quad (1.22)$$

Similar to the Metropolis algorithm, the greatest acceptance ratio is chosen to be 1. If $HS_i > 0$, the exponential is smaller than unity and $A(b,a)$ is set to 1 with $A(a,b) = A = e^{-2\beta kHS_i}$. If $HS_i \leq 0$, the exponential is greater or equal to unity and $A(a,b)$ is set to 1. Lastly, $A = e^{-2\beta kHS_i} = 1$ if there is no magnetic field.

Again, what this means is if $H = 0$, the acceptance ratio is unity and once a cluster is constructed it is always flipped. If $HS_i < 0$, the acceptance ratio is also unity. If $HS_i > 0$, the cluster is flipped with probability $e^{-2\beta kHS_i}$.

The application of the Wolff algorithm for the Ising model can be summarized as such:

1. Pick a random lattice site as a seed spin which begins the construction of a virtual cluster.
2. Look at all its nearest neighbors. If a neighbor has the same spin value as the seed spin, it can be added to the virtual cluster with a probability $P_J = 1 - e^{-2\beta J}$.
3. When a spin is added to the cluster, also consider all of its neighbors. There can be multiple opportunities for spins to be added to the cluster due to the way the cluster is constructed.
4. Continue until there are no more spins to add. Generate a random number $0 \leq r < 1$ and calculate the acceptance ratio A . If $r < A$, flip the entirety of the virtual

cluster.

The algorithm satisfies ergodicity since it is possible to have single-spin clusters, which allows any state transition given enough time. The linking probability P_J can be precalculated, as it depends only on J . The acceptance ratio $A(a, b)$ can also be calculated in advance, but it depends on k , which can take any value from 1 to N .

In the following applications, it will be seen the Wolff algorithm is particularly useful in the study of granular systems.

1.5 Monte Carlo Simulations of the Ising Model

The 1D Ising model was solved exactly by Ising and was found to exhibit no phase transition at $T > 0$. The 2D square lattice Ising model, solved by Onsager [27], does admit a second order (continuous) phase transition at the critical temperature $T_c = \frac{2}{\ln(1+\sqrt{2})}J \approx 2.269$ for $H = 0$, $J = 1$. A plot of the magnetization against temperature is shown in Fig. 1.2 for an infinite lattice, where it can be seen that the order parameter (the magnetization) becomes 0 at T_c . There is no discontinuity in the magnetization, which illustrates that the phase transition is continuous.

Since simulations cannot be done on infinite lattices, the 2D model involves $N = L \times L$ spins [24]. Of course, not having an infinite lattice will give rise to finite size effects. These effects have been studied extensively and lead to a shift in location and shape of the peaks in the susceptibility and heat capacity. The exact solution of the finite size Ising model allows comparisons with the magnetization and other quantities to be obtained for finite L , which gives an indication of the validity of simulations.

The effects of having a finite lattice can be reduced by a judicious choice of boundary conditions. Periodic boundary conditions involve assigning these missing neighbor connections in a periodic fashions, so that spin 1 and spin L on a line are connected. As such, every single spin on a square lattice has exactly four nearest neighbors.

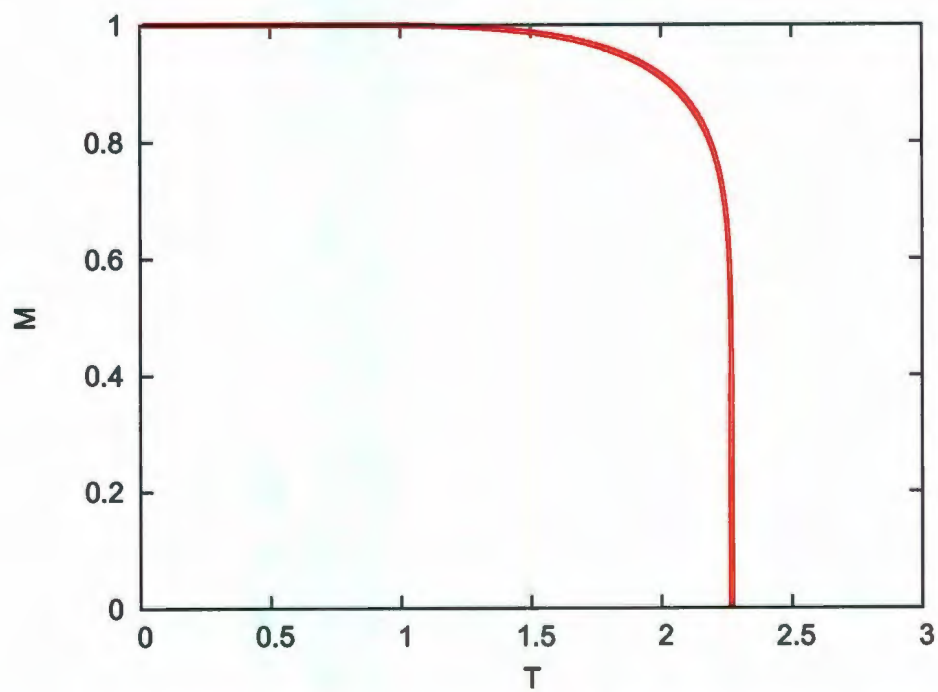


Figure 1.2: Onsager exact solution for the magnetization vs. temperature of the 2D Ising model.

1.6 Monte Carlo Step

A Metropolis Monte Carlo step (MCS) is defined as one sweep of the lattice with the Metropolis algorithm. A single sweep of the lattice can be done in two different ways, by either sequentially going over each N spins of the lattice or by selecting a random lattice site N times. In the second case, there will be spins which will be picked more than once in a single MCS, but it removes the artificial ordering of spin flipping that sequentially picking spins introduces. As such, the second case is more often used. A Wolff MCS is defined as building one virtual cluster and trying to flip it. Unless otherwise stated, the work done uses a hybrid Monte Carlo scheme where a MCS is defined by one Metropolis MCS followed by one Wolff MCS. For equilibration reasons, a certain amount of initial steps MCS_0 are discarded, which is usually taken to be 10% of the total number of MCS. MCS_U is defined as the total amount of useful steps from which quantities are calculated; thus, a run that has 10000 MCS and 1000 MCS_0 has $MCS_U = 9000$ useful steps. MCS_T is defined as the total amount of steps used, so that $MCS_T = MCS_0 + MCS_U$. The MCS_T used in the present study have a typical value of 50000, as much longer requires too long to run and much lower is not typically long enough to reach equilibrium. Looking at the behavior of a quantity of interest as a function of MCS_T for one specific temperature gives an idea of when equilibration is reached for that temperature, while averaging over several runs can be done to ensure a more accurate result.

1.7 Thermodynamic quantities

To draw meaning from the results, there are a few quantities already mentioned that are calculated: the magnetization, the energy, the specific heat and the susceptibility. The magnetization and the energy of the system are calculated after every MCS after MCS_0 steps have passed. An average over the MCS_U steps is taken for these quantities because of the randomness of the process. From these, several results are calculated, with $\langle X \rangle$ indicating an average over MCS_U steps.

The magnetization per spin, referred to as the total magnetization or simply magnetization, in contrast with the grain magnetization later introduced, is defined through

$$M(T) = \frac{1}{N} \left\langle \left| \sum_i S_i \right| \right\rangle, \quad (1.23)$$

where i is summed over all N lattice sites. In addition, the magnetic susceptibility per spin was calculated using:

$$\chi(T) = \beta N (\langle M(T)^2 \rangle - \langle M(T) \rangle^2), \quad (1.24)$$

where $\beta = 1/kT$ and the specific heat per spin is given by

$$C(T) = \frac{\beta^2}{N} (\langle E(T)^2 \rangle - \langle E(T) \rangle^2). \quad (1.25)$$

For the purposes of Monte Carlo simulations, $k \equiv 1$. While the magnetization can be intuitively defined, the susceptibility and specific heat follow a less straightforward derivation.

Energy and Specific Heat

From the partition function in Eq. 1.7:

$$\frac{\partial Z}{\partial \beta} = - \sum_a E_a e^{-\beta E_a}, \quad (1.26)$$

and so, with Eq. 1.8

$$-\frac{1}{Z} \frac{\partial Z}{\partial \beta} = E = -\frac{\partial \log Z}{\partial \beta}, \quad (1.27)$$

where Q was replaced by the total energy of the system E and Q_a by E_a . The specific heat of the system is given by

$$C \equiv \frac{\partial E}{\partial T} = -k\beta^2 \frac{\partial E}{\partial \beta} = k\beta^2 \frac{\partial^2 \log Z}{\partial \beta^2}. \quad (1.28)$$

While the quantities Q themselves provide important information on the system, their standard deviation can also be looked at. The variance of Q is

$$\langle (Q - \langle Q \rangle)^2 \rangle = \langle Q^2 \rangle - \langle Q \rangle^2. \quad (1.29)$$

From Eq. 1.8 and using Eq. 1.26, $\langle E^2 \rangle$ is given by

$$\langle E^2 \rangle = \frac{1}{Z} \sum_a E_a^2 e^{-\beta E_a} = \frac{1}{Z} \frac{\partial^2 Z}{\partial \beta^2}. \quad (1.30)$$

Thus,

$$\langle E^2 \rangle - \langle E \rangle^2 = \frac{1}{Z} \frac{\partial^2 Z}{\partial \beta^2} - \left(-\frac{1}{Z} \frac{\partial Z}{\partial \beta} \right)^2 = \frac{\partial^2 \log Z}{\partial \beta^2}. \quad (1.31)$$

Finally, the system specific heat can be equated to the variation of the energy

$$\frac{C}{k\beta^2} = \langle E^2 \rangle - \langle E \rangle^2, \quad (1.32)$$

$$C = k\beta^2 (\langle E^2 \rangle - \langle E \rangle^2). \quad (1.33)$$

The specific heat often used for Monte Carlo simulations is calculated per spin, as mentioned in Sec. 1.7, and a factor of $1/N$ is introduced.

Magnetization and Susceptibility

To measure the response of the system magnetization M to a change in magnetic field, the system magnetic susceptibility is defined by

$$\chi = \frac{\partial \langle M \rangle}{\partial H}, \quad (1.34)$$

where $\langle M \rangle$ is the average magnetization and H is the external magnetic field. The average magnetization and average magnetization squared are given by

$$\langle M \rangle = \frac{\sum_a M_a e^{-\beta E_a}}{Z}, \quad (1.35)$$

$$\langle M^2 \rangle = \frac{\sum_a M_a^2 e^{-\beta E_a}}{Z}. \quad (1.36)$$

The effect of an external magnetic field on magnets is given by the Zeeman term, defined by, for example, the last term in Eq. 1.2 which comes down to, for a state a , $-HM_a$. Since this term is only in the energy E_a ,

$$\frac{\partial Z}{\partial H} = \beta \sum_a M_a e^{-\beta E_a}. \quad (1.37)$$

The susceptibility becomes

$$\chi = \frac{\partial \langle M \rangle}{\partial H} = \frac{\frac{\partial \sum_a M_a e^{-\beta E_a}}{\partial H} Z - \sum_a M_a e^{-\beta E_a} \frac{\partial Z}{\partial H}}{Z^2}. \quad (1.38)$$

Finally, the susceptibility is given as

$$\chi = \frac{\beta \sum_a M_a^2 e^{-\beta E_a} Z - \beta (\sum_a M_a e^{-\beta E_a})^2}{Z^2} = \beta (\langle M^2 \rangle - \langle M \rangle^2), \quad (1.39)$$

which is similar to specific heat, as it involves the variance. The susceptibility often used for Monte Carlo simulations is calculated per spin, as mentioned in Sec. 1.7, and a factor of N is introduced when using the magnetization per spin.

Chapter 2

Ising Model of Grains

2.1 Granular Ising Model

Within magnetic recording media, bits are made of multiple grains that typically have length scales of the order of several nanometers. In micromagnetics, these grains are assumed to have a uniform magnetization. As the grains get smaller and smaller, to obtain denser information storage, the effects and validity of this assumption needs to be carefully examined.

Monte Carlo simulations have been used to study the Ising model for a variety of lattice structures in two and three dimensions and for different types and combinations of interactions, where exact solutions are not possible. Such variations of the basic Ising model can be used to study a wide variety of magnetic systems of physical interest, such as antiferromagnets, spin glasses and frustrated systems.

In this thesis, an Ising model constructed to explicitly take into account the effect of inter-grain (labelled J) and intra-grain (labelled J') exchange interactions is studied. The use of the near-neighbor Ising model is a limit of the anisotropic Heisenberg model, studied in Chap. 4, which more accurately describes magnetic recording media. These materials are often cobalt-based and have a high uniaxial anisotropy that can be approximated by the Ising model. Single-layer 2D models and multilayer models, which

emulate thin films, are constructed with periodic boundary conditions and separated into grains, with J and J' as nearest neighbor exchange interactions for spins between grains and spins inside of grains, respectively. Fig. 2.1 shows a schematic of the 2D lattice, with grains represented as islands.

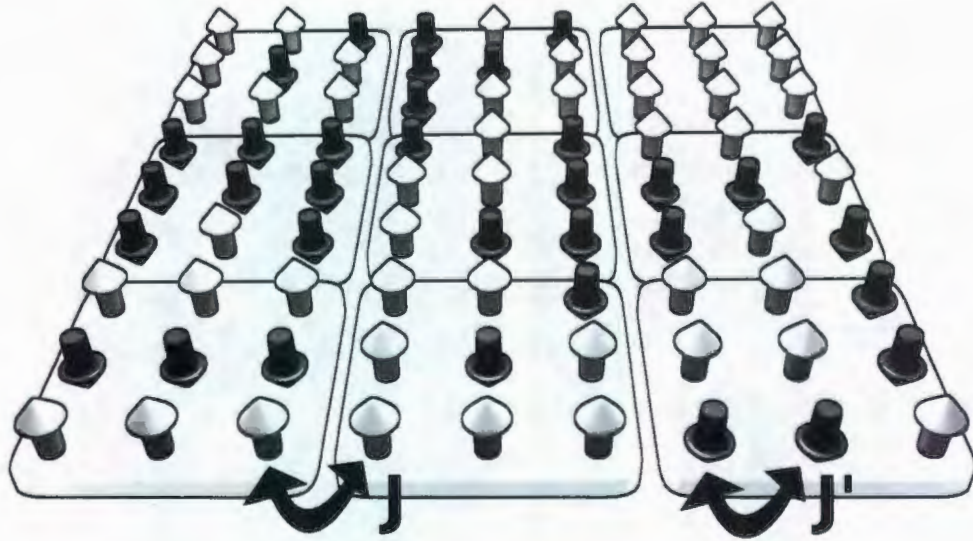


Figure 2.1: Schematic of the 2D model with 3×3 grains each having 3×3 spins. The labels of J and J' on the figure show locations where these nearest neighbor interactions apply.

The grains are modelled to have L' spins in the x and y directions and z spins in the z direction. As such, a grain has a total of $L' \times L' \times z$ spins, and the total system has $L \times L$ grains, a total of $N = (LL')^2 z$ spins.

In the multilayered case ($z > 1$), the top and bottom layers of the model have open boundary conditions to better represent the thin films of magnetic recording media.

The magnetization of a single grain is defined as

$$M_I = \frac{1}{N_g} \left| \sum_{j \in I} S_j \right|, \quad (2.1)$$

where $\sum_{j \in I}$ denotes the sum over all $L' \times L' \times z$ spins in grain I . The thermal average over all grain magnetizations is then given by

$$M_g(T) = \frac{1}{L^2} \left\langle \sum_I M_I \right\rangle. \quad (2.2)$$

To calculate the effects of the granular characteristic of the films, the difference between J and J' needs to be taken into account. The only effect of a varying J_{ij} to the Metropolis algorithm is in the evaluation of ΔE . As J_{ij} is only used when calculating the nearest neighbor energy, only Eq. 1.15 changes, depending on whether or not the nearest neighbor is connected with J or J' . This in turn changes Table 1.1, as the number of choices increases significantly. The new lookup table then has many more elements, but they can still easily be discretized.

The derivation of the effect on the Wolff algorithm is a bit more involved and involves changing the linking probability. In addition to the m and n bonds from the derivation in Sec. 1.4.6, there are also m' and n' bonds defined analogously to m and n to be taken in consideration. At the cluster boundary, there are four possible near-neighbor spin-spin interactions: J between two same-valued spins across grain boundaries (of which there are m), J between two different-valued spins (n), J' between two same-valued spins within a grain (m') and J' between two different valued spins (n'). As such, $g(a, b)$ becomes $g_0(1 - P_J)^m(1 - P_{J'})^{m'}$ and $g(b, a)$ becomes $g_0(1 - P_J)^n(1 - P_{J'})^{n'}$ where $P_{J'}$ is the probability of adding a spin to the cluster if the bond under consideration is linked with J' instead of J . Again, the g_0 is the same for the forward and the backward move and can be a complicated factor that depends on the cluster and lattice size. Detailed balance is then given by

$$\frac{P(a, b)}{P(b, a)} = \frac{g_0(1 - P_J)^m(1 - P_{J'})^{m'} A(a, b)}{g_0(1 - P_J)^n(1 - P_{J'})^{n'} A(b, a)} = e^{-\beta \Delta E}. \quad (2.3)$$

ΔE can be split into different parts, such that $\Delta E = \Delta E_H + \Delta E_J + \Delta E_{J'}$. Compared

to the non-granular model, $\Delta E_H = 2kHS_i$ and $\Delta E_J = 2J(m - n)$ do not change. $\Delta E_{J'}$, derived in a very similar way to ΔE_J , is $2J'(m' - n')$. The ratio of acceptance probabilities then becomes

$$\begin{aligned} \frac{A(a, b)}{A(b, a)} &= (1 - P_J)^{-(m-n)} (1 - P_{J'})^{-(m'-n')} e^{-\beta(2J(m-n))} e^{-\beta(2J'(m'-n'))} e^{-2\beta kHS_i} \\ &= (e^{2\beta J}(1 - P_J))^{-(m-n)} (e^{2\beta J'}(1 - P_{J'}))^{-(m'-n')} e^{-2\beta kHS_i}. \end{aligned} \quad (2.4)$$

In a very similar fashion to the homogeneous Ising model, P_J can be chosen to be $1 - e^{-2\beta J}$ and $P_{J'}$ can be chosen to be $1 - e^{-2\beta J'}$. The algorithm is carried out in the same way as described earlier for the uniform Ising model except that when considering whether the spins are added to the cluster, P_J is used if the spin bond uses J while $P_{J'}$ is used for a bond connected with J' .

2.2 Simulation of Granular Media

2.2.1 Initial simulations

Ideally, L should be chosen such that finite size effects are negligible. Unfortunately, this is not feasible as it would require L to be immensely big. Fig. 2.2 shows the effect of changing the lattice size on the magnetization, the specific heat and the susceptibility vs. temperature for a homogeneous ($J = J' = 1$) 2D Ising model using the Metropolis algorithm.

The critical temperature T_c can be obtained from the simulation data by a variety of methods: the temperature at which the magnetization extrapolates to zero, the temperature at which the susceptibility shows a large peak and the temperature at which the specific heat also shows its peak. The value obtained will depend on L to a certain extent, which will add a systematic error to T_c ; however, extracting T_c from more than one method and extrapolating to higher values of L helps to diminish this error. Finite-

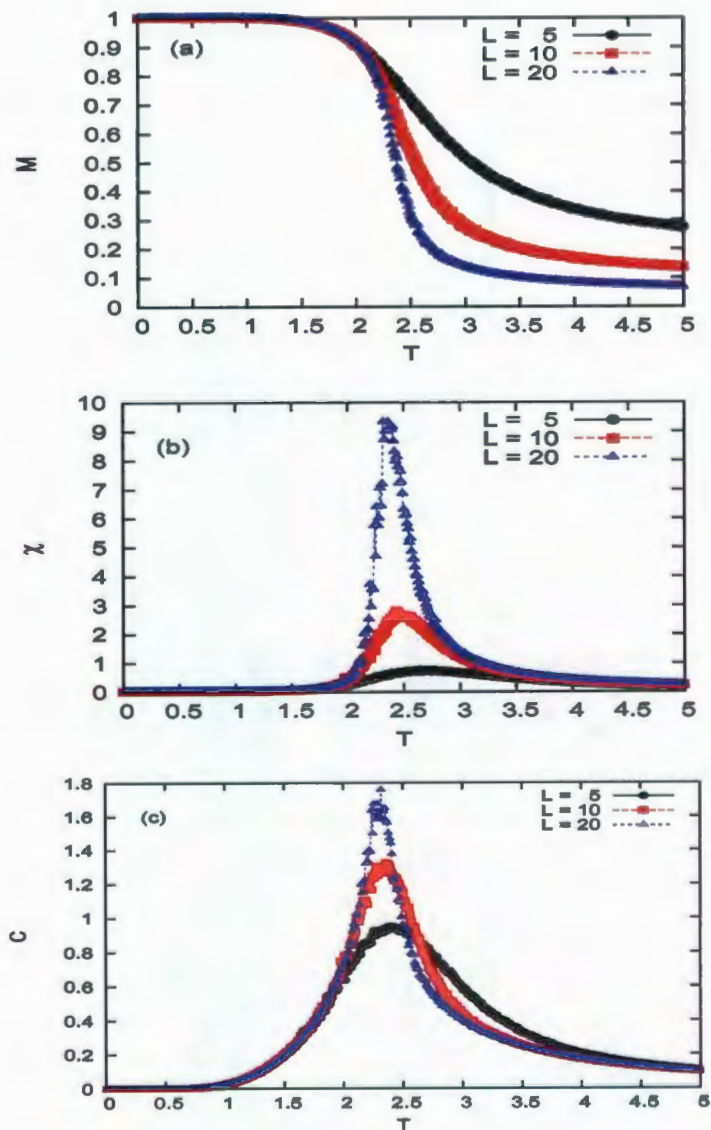


Figure 2.2: Magnetization (a), susceptibility (b) and specific heat (c) vs. temperature for a homogeneous (having no grains) 2D Ising model with the Metropolis algorithm with different values of the lattice size (shown in the legend). Here, $MCS_T = 50000$.

size scaling techniques can be used to obtain a more accurate value in some cases, but for the current work the present methods are adequate.

In the case of the granular 2D square-lattice Ising model, the situation is somewhat

more complicated as there are two different length scales to consider. Here, L' must be sufficiently large as it represents a mini-Ising model within each grain. L must also be large enough to capture the essential system features. Due to computational reasons, it would be exorbitantly expensive to have much more than even $L = L' = 20$, as that would entail 160000 spins for every 2D layer which is not feasible, except perhaps in the single-layer case ($z = 1$). As such, L and L' are typically chosen to be smaller while still giving the simulation of the relevant physics. In all of the following results, $J' \equiv 1$.

Shown in Fig. 2.3 is a graph of the magnetization vs. temperature with the granular model using the Metropolis algorithm for a small system size ($L = 4$, $L' = 5$) for finite and zero J .

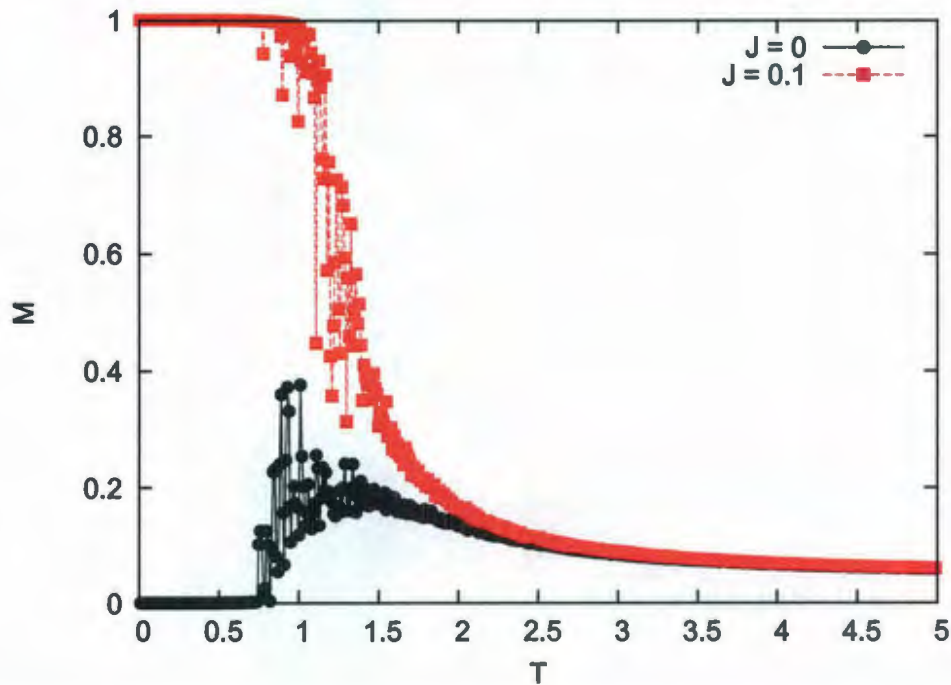


Figure 2.3: Magnetization vs. temperature of the granular Ising model using only the Metropolis algorithm for $J = 0$ and $J = 0.1$. Large fluctuations are seen near the critical temperature. Here, $MCS_T = 50000$.

As can be seen, there are some equilibration issues even when using a reasonably

large value of MCS_T . While the data is obtained by starting at the highest temperature and lowering it, the $J = 0$ case shows an unexpected behavior, falling to zero near $T = 0$. This is due to the choice of algorithm, as later results do not show this behavior. When $J \neq 0$, there are two different length scales to the problem. While the spins inside a grain can equilibrate about as well as they would be able to if the simulation was done on a small homogeneous system, the inter-grain correlations are not able to equilibrate. To flip an entire grain, the Metropolis algorithm has to go through every spin from the grain and flip them. At lower temperatures, the Metropolis algorithm has diminished chances to flip spins which makes it even more difficult for the system to equilibrate.

As equilibrium is required to obtain results of the critical temperatures T_c , different simulation techniques were explored to equilibrate the system. A possible solution would be to simply raise MCS_T ; however, this suggestion has the disadvantage of not being able to scale well nor does it guarantee good results for values accessible in a reasonable amount of computer time. Shown in Fig. 2.4 is the situation for the interactions of a single grain with its neighbors. The spins at the edge of the grain are the ones that facilitate the grain flipping, but a single spin could flip many times before helping its neighbors flip. The single spin at the edge on the right that was flipped has three different neighbors coupled with a strong J' interaction that favors the spin flipping back to match its own grain's spin direction.

For larger lattices, the amount of computer time needed to keep a high value of MCS_T increases very quickly. For the 2D model, the amount of calculations required is $O(L^2)$ since a linear increase in L quadratically adds to the amount of spins. If the grains get bigger, the effect seen in Fig. 2.4 becomes even more pronounced since there are even more chances for the progress of the grain flipping to be hampered. It is therefore difficult to equilibrate the system using a single-spin-flip algorithm such as the Metropolis algorithm.

The Wolff cluster-flipping algorithm explained in Sec. 1.4.6 provides a more efficient alternative to the Metropolis algorithm. The Wolff algorithm, originally designed to help

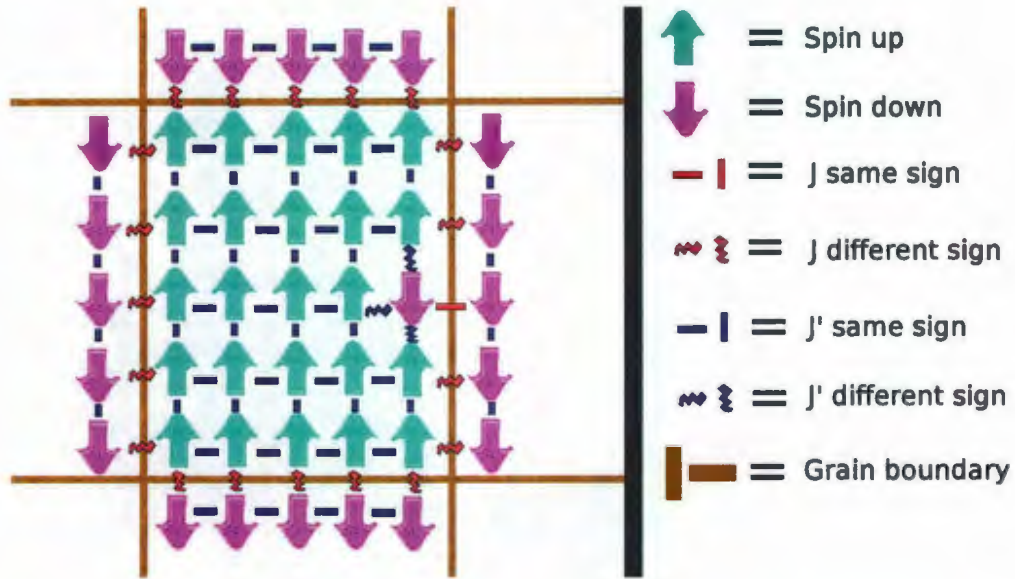


Figure 2.4: Representation of a single grain showing the effort needed to flip it when using only the Metropolis algorithm. After a single spin is flipped on the right side of the grain, the system might be more inclined to flip it again rather than flip the other spins in the grain.

with critical slowing down, has the advantage of allowing multiple spins to be flipped simultaneously and as such allows equilibrium to be reached very quickly. This makes it a good choice for what is seen in Fig. 2.3, where there are large fluctuations of the magnetization around the critical temperature when grains are added to the model.

While the Wolff algorithm alone does satisfy ergodicity and detailed balance, it would take many iterations to sweep over the entire lattice when the spins are highly disordered, such as at high temperatures. Combining the Metropolis and Wolff algorithm results in a method that is efficient at both high and low temperature. A single MCS is then defined to be one Metropolis sweep of the lattice along with one Wolff cluster, whether it flips or not. This hybridization has been studied in the past [28] and proves to be very effective for the simple Ising model and it will be shown that it also works well in the model for granular media considered in the present work.

2.2.2 2D Single-Layer Ising Model Results

The first simulations for this model were done on a single-layer Ising model to test its validity before being extended to the multilayer case. The total magnetization is plotted in Fig. 2.5 for different values of J , the inter-grain coupling. In recording media, it is estimated that J is 10-100 times smaller than J' .

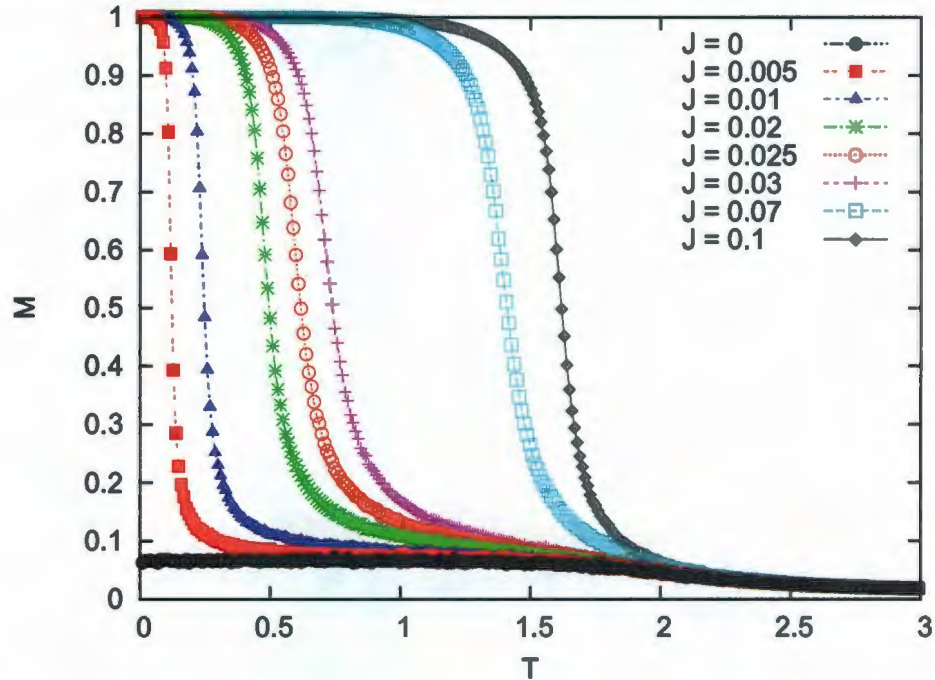


Figure 2.5: Total magnetization vs. temperature in the 2D case with $L = 12$ and $L' = 10$ as a function of inter-grain coupling J (values shown in the legend).

It can immediately be seen that the fluctuations in the magnetization calculated from the Wolff algorithm are much less than those calculated using the simple Metropolis algorithm, especially around the critical temperature.

The $J = 0$ case shows a different behavior from Fig. 2.3, settling at around 0.07 when $T = 0$. The magnetization at $T = 0$ when $J = 0$ can be estimated by noting that the grains are completely independent and uniformly magnetized

$$M(N_g) = \sum_{i=0}^{N_g} M'(i)P(i) = \frac{1}{N_g 2^{N_g}} \sum_{i=0}^{N_g} |N_g - 2i| \frac{N_g!}{(N_g - i)!i!}, \quad (2.5)$$

where N_g is the number of grains, $M'(i) = \frac{1}{N_g} |N_g - 2i|$ is the magnetization when there are i grains pointing down and $P(i) = \frac{1}{2^{N_g}} \frac{N_g!}{(N_g - i)!i!}$ covers the number of ways of selecting i grains pointing down (normalized binomial coefficient in factorial form). As an example, if there were 2 grains, $N_g = 2$ and the sum $\sum_{i=0}^2$ accounts for the situations of zero grains pointing down, one grain pointing down and two grains pointing down. If one grain is pointing down, $i = 1$, $M'(1) = 0$ and $P(1) = 0.5$, as half of the configurations occur when only one grain is pointing down. If either both grains are pointing down or both grains are pointing up, $M'(i) = 1$ and $P(i) = 0.25$. The sum is then $M(N_g = 2) = M'(0)P(0) + M'(1)P(1) + M'(2)P(2) = 0.25 + 0 + 0.25 = 0.5$. For $N_g = 144$, Eq. 2.5 gives 0.0664 which is consistent with Fig. 2.3, while with an infinite system, one can expect $M(N_g \rightarrow \infty) = 0$ as the finite size effect is removed. The earlier use of the Metropolis algorithm in Fig. 2.3 showed erratic behavior as the temperature dropped down, but using the Wolff algorithm gives the expected result, though a much higher MCS_T would allow the Metropolis algorithm to eventually achieve the correct result. This shows that using the Wolff algorithm allows a quicker method of reaching equilibrium.

Fig. 2.6 shows the grain magnetization (defined in Sec. 2.1) obtained by simulation. The grain magnetization has a shape vaguely similar to the total magnetization and has a drop around $T = 2$. As L' is only 5 or 10, what looks like a finite size effect is seen, with a sharper drop for the higher L' . It is also seen in these results, as opposed to the total magnetization, that J has very little effect on the grain magnetization.

Fig. 2.7 shows the susceptibility χ (defined in Sec. 1.7) obtained in the same simulation as the total magnetization. Similar to the homogeneous model, the peak in the susceptibility provides a good indication on the location of the phase transition associated with T_c , the ordering between the grains. While the estimates of T_c calculated from

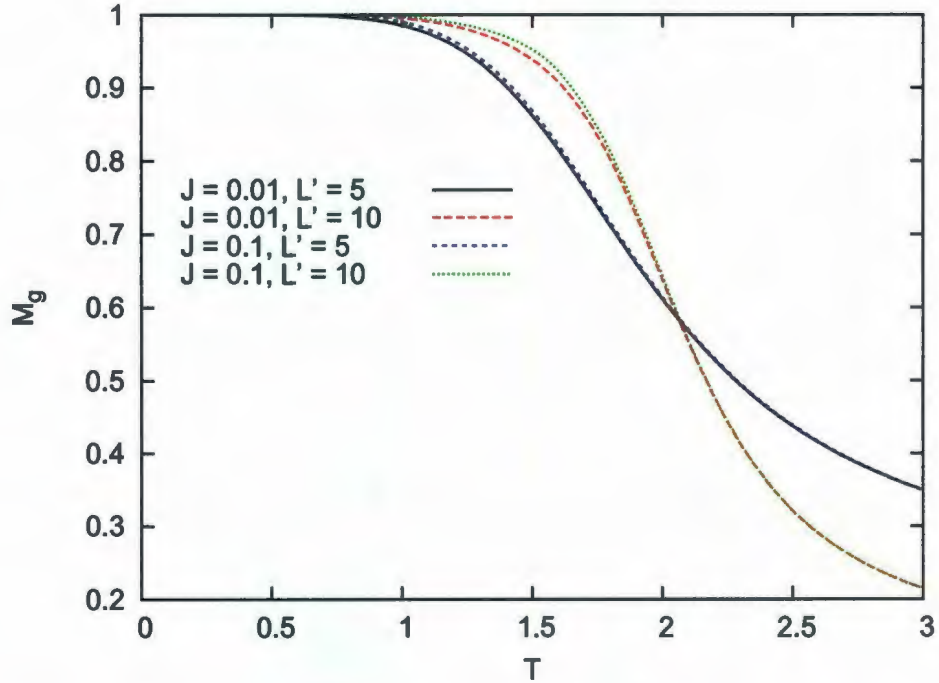


Figure 2.6: Grain magnetization vs. temperature in the 2D case with $L = 12$ for two grain sizes $L' = 5$ and $L' = 10$ and two values of $J = 0.01$ and $J = 0.1$ as shown in the legend.

the magnetization data are consistent with those obtained from the susceptibility, the susceptibility often provides a slightly more accurate method to obtain T_c .

Fig. 2.8 shows the specific heat C (defined in Sec. 1.7) obtained in the same simulation as the other quantities. The specific heat for the granular model, much like the one for the homogeneous model, has a very distinct peak around a temperature that is defined here as T'_c .

There are several important points to note about T'_c . Firstly, it occurs above the transition temperature T_c obtained from the magnetization/susceptibility data. Secondly, it corresponds loosely to the temperature at which grain magnetization drops rapidly. Finally, unlike T_c , it is almost independent of the value of J . A close examination of the data shown in Fig. 2.8 also reveals some secondary peaks. The location of these peaks depends on the value of J and corresponds to the peaks in the magnetic susceptibility

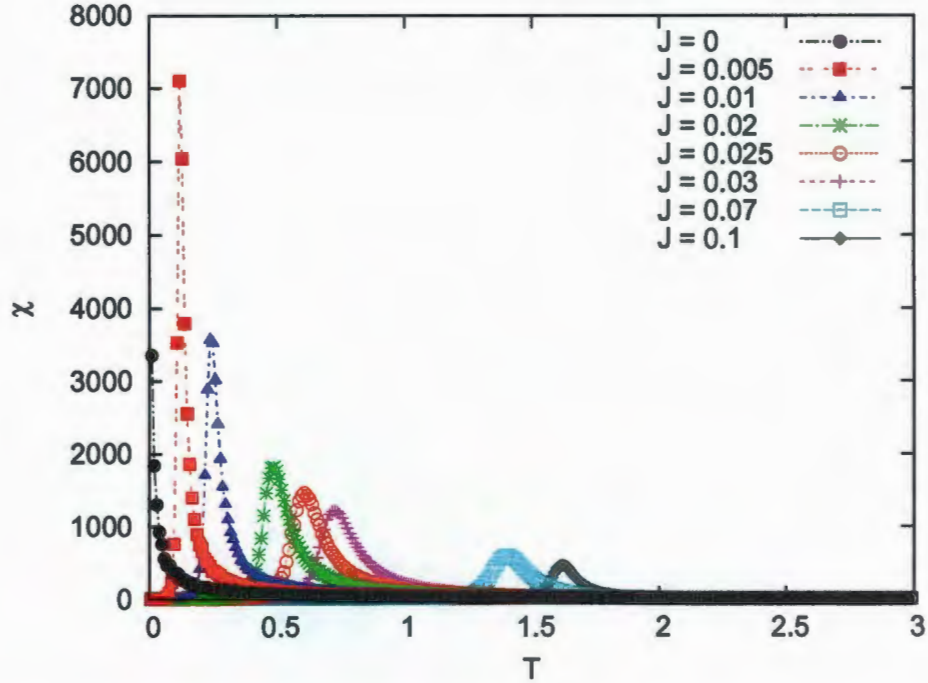


Figure 2.7: Susceptibility vs. temperature in the 2D case with $L = 12$ and $L' = 10$ as a function of inter-grain coupling J (values shown in the legend).

of T_c .

Another way to draw meaning from the results is by looking at the spin configurations at specific points of the simulation. Fig. 2.9 shows one spin configuration for the 2D Ising model at a specific point in the simulation. The spin configuration shows the spins to be highly ordered within the grain, while the grains themselves show a considerable degree of disorder. In effect, the temperature of the system at that time is below T'_c and above T_c . When the temperature of the system is much higher than both T_c and T'_c , all the spins of the lattice are disordered. As the temperature lowers to reach T'_c , the spins inside grains align to become correlated. This is what is seen in Fig. 2.9, where the grains are “mostly black” or “mostly white”. The grains, however, remain uncorrelated such that a “mostly black” grain might or might not have a “mostly black” neighbor. As the temperature lowers even more to reach T_c , the grains themselves order and the

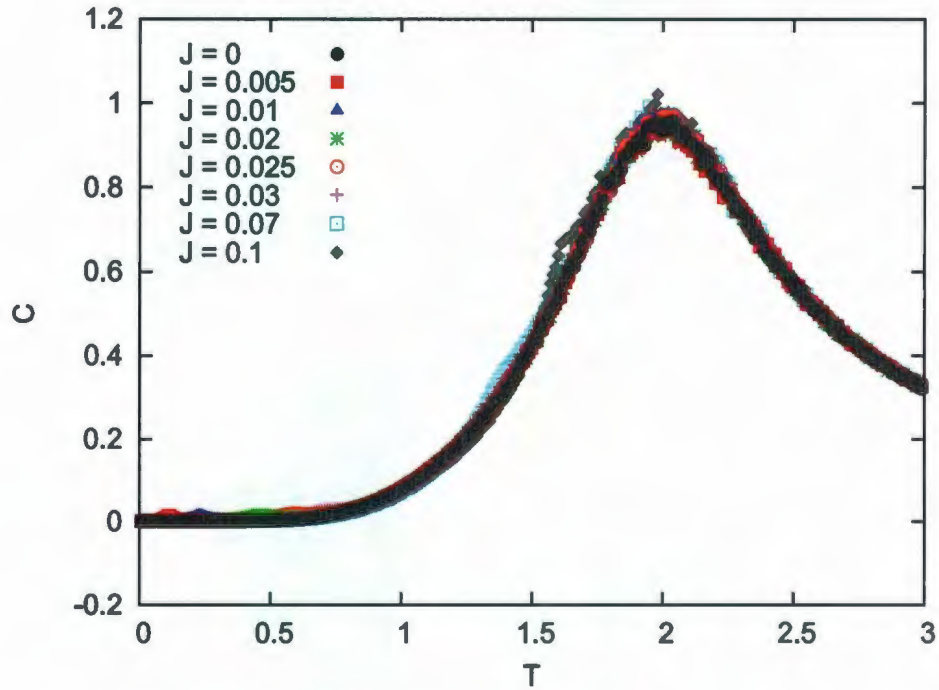


Figure 2.8: Heat capacity vs. temperature in the 2D case with $L = 12$ and $L' = 10$ as a function of inter-grain coupling J (values shown in the legend).

entire lattice has a preferred spin direction. It is seen in the snapshot that the different grains are mostly uncorrelated, as expected for $T > T_c$. This is reflected in the total magnetization graph, where above T_c the system magnetization is effectively very small.

These various quantities provide different approaches to draw results from the simulation, but in order to relate to the initial hypothesis of the effect of approximating grains as a single spin, some assumptions must be probed. If a single grain can safely be treated as uniformly magnetized, it is expected that approximating grains by single spins in this model would be analogous to the homogeneous model, with single lattice sites interacting with neighbors with some effective exchange constant of $\propto J$. By plotting T_c in the granular model for different values of the exchange constant J , the linear relationship between the effective exchange constant and J can be tested.

Fig. 2.10 shows T_c vs. J obtained from Figs. 2.5 - 2.8 with $L' = 5$ and $L' = 10$.

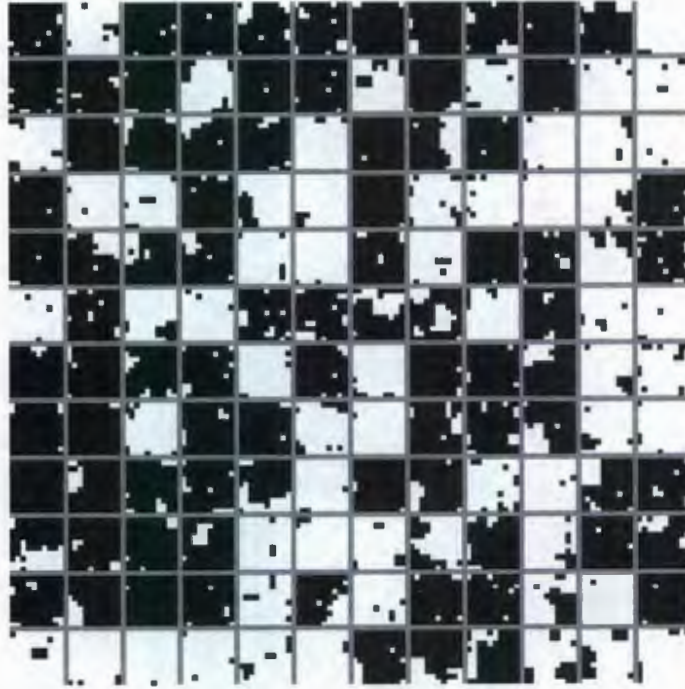


Figure 2.9: Spin configuration in the 2D case with $L = 12$ and $L' = 10$ using $J = 0.03$ at $T = 1.79$ for $T_c' \approx 2.0$ and $T_c \approx 0.6$. Up and down spins are shown as black and white points and grain boundaries as grey lines.

For both cases, with low values of J , T_c shows a linear dependance on J , but for larger values of J the transition temperature deviates significantly from linear behavior. The effect is even more pronounced for the higher L' case, with a deviation starting at a lower value of J . The slope of the lines for $L' = 5$ and $L' = 10$ were respectively 11.4 and 23.37. These values are not, in fact, 2.269. To explain this behavior, a new quantity J_{eff} is introduced as

$$J_{\text{eff}} = JL'. \quad (2.6)$$

This quantity allows a description of the exchange interaction J while being independent

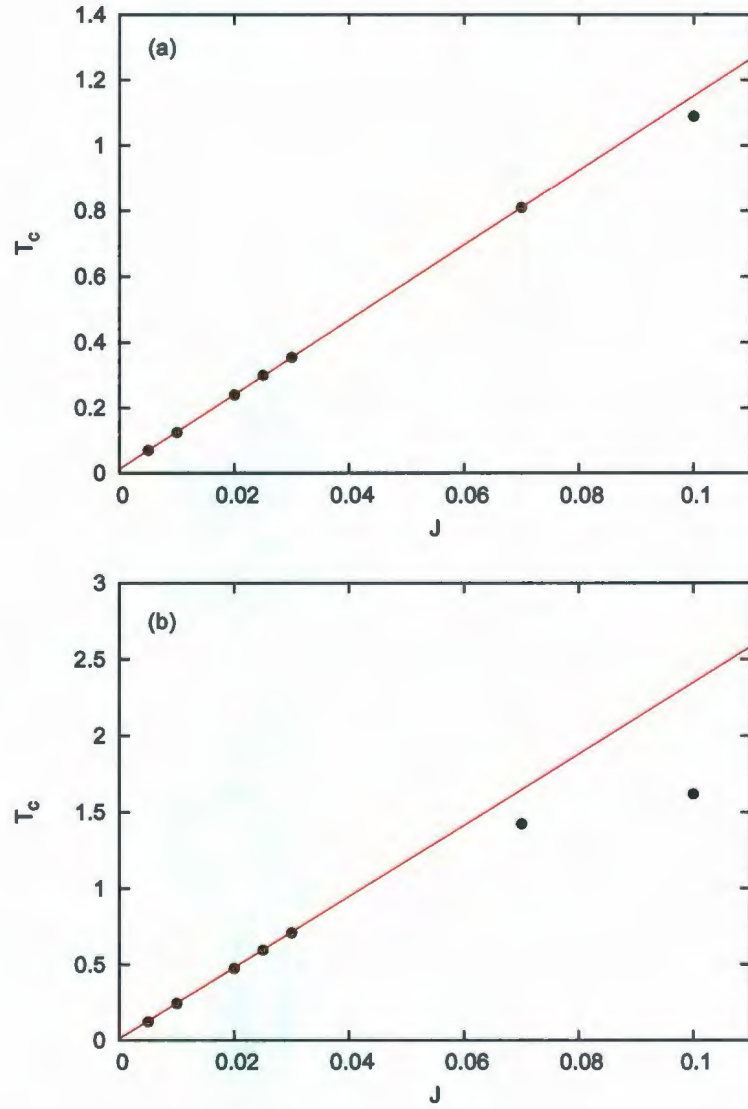


Figure 2.10: T_c vs. J in the 2D case with $L = 12$ and $L' = 5$ for Fig. (a) and $L' = 10$ for Fig. (b).

of L' . In micromagnetics, this can be taken into account by the exchange interactions scaling according to the cross-sectional area between grains. It is then seen that the slope of the linear section is in fact close to $2.269J_{\text{eff}}$. These results thus suggest that J

times the number of spins along a side of the grain gives an effective measure of inter-granular exchange in the single-layer case. They also show that there can be significant deviations from a linear relation between T_c and J . To better understand this effect, simulations were undertaken on a multi-layer thin film model.

2.2.3 Results for Multilayer Systems

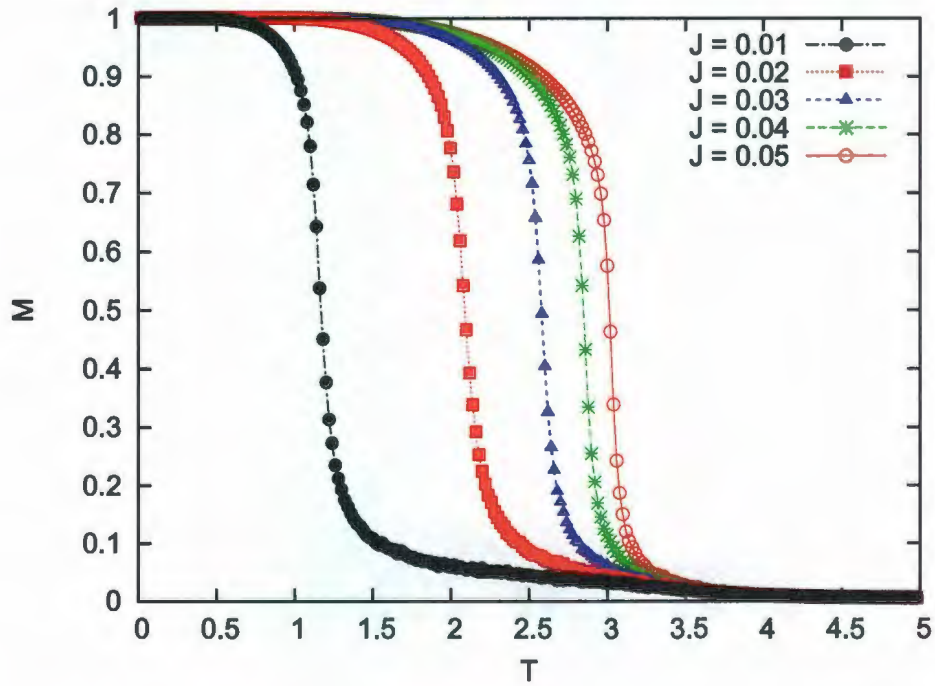


Figure 2.11: Magnetization vs. temperature for different values of J shown in the legend. Here, $L' = 10$ and $z = 10$.

Similar to the single layer case, examples of results for the total magnetization, susceptibility and specific heat used to estimate T'_c and T_c for the multilayer ($z > 1$) granular Ising model are plotted in Figs. 2.11 - 2.13. Within the accuracy of the simulations, the values of T'_c are insensitive to the specific value of J , depending only on the variables z and L' .

Just as in the single layer case, the critical temperature of the grains T_c can be

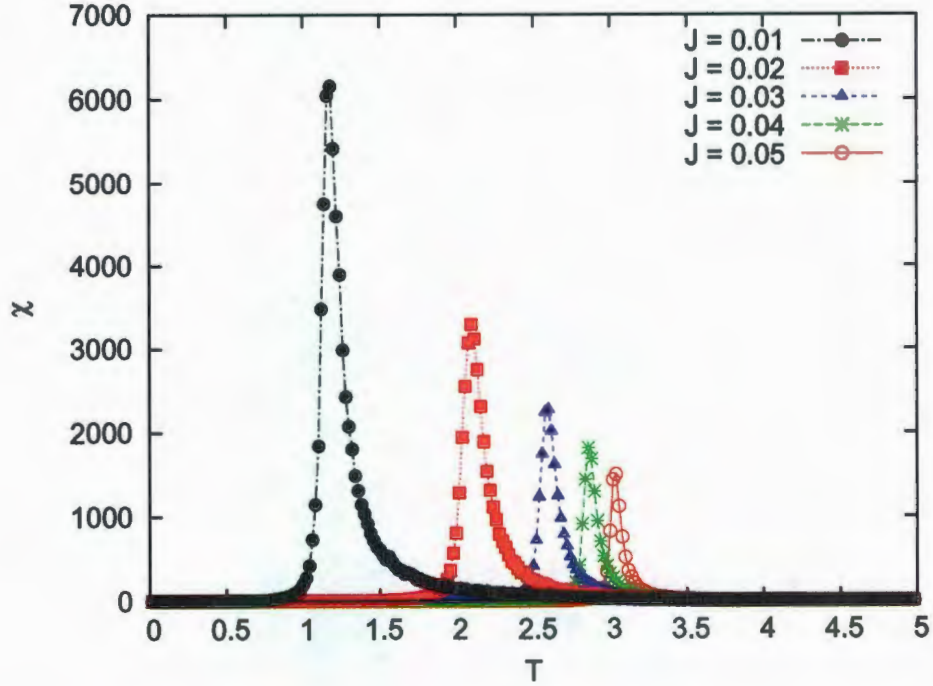


Figure 2.12: Susceptibility vs. temperature for different values of J shown in the legend. Here, $L' = 10$ and $z = 10$.

plotted as a function of J to observe what is expected to be a linear regime. It was noted in Eq. 2.6 that an effective J is used that takes into account the amount of spins on one side of the grain, L' . For the multilayer model, the number of spins in a grain is $L'^2 z$. If a grain is to be approximated by a single spin, this entails that the multilayer system reduces to a simple single layer model with each grain represented by a single spin. The spins denoting the single grains will couple through an effective exchange constant which was labeled J_{eff} , such that $J_{\text{eff}} \propto J$. For the single-layer model, the constant of proportionality is L' , the amount of spins on one side of the grain; this is the number of spins for a single grain that interact with its nearest grain neighbor. For the multilayer model, the number of spins of a single grain that interact with one neighbor grain is now $L'z$. This situation is shown in Fig. 2.14.

As such, for the multilayer model J_{eff} is defined as

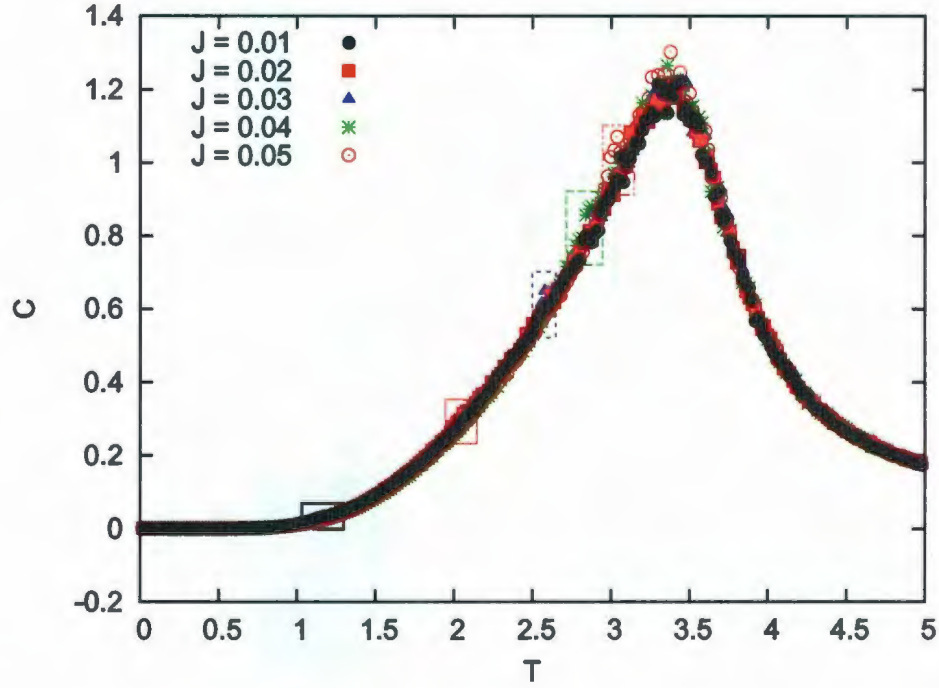


Figure 2.13: Heat capacity vs. temperature for different values of J shown in the legend. Here, $L' = 10$ and $z = 10$. The boxed area approximate the location of the peaks.

$$J_{\text{eff}} = JL'z. \quad (2.7)$$

The critical temperature of the grains T_c can then be plotted against J_{eff} to test this conjecture. Fig. 2.15 shows this for a few different simulation parameters. Again, with low values of J , T_c shows a linear dependence on J , but for larger values of J the transition temperature deviates significantly from linear behavior. The slope of the linear regime is 2.27 which is very close to the expected result of $T_c \approx 2.269J_{\text{eff}}$ of the 2D Ising model. This result confirms what was observed for the $z = 1$ case.

While the 2D model has a critical temperature $T_c \approx 2.269$ which can be determined exactly, the full 3D model cannot be solved, but the critical temperature has been estimated by Monte Carlo simulations [29] to be $T_c \approx 4.512J$. This implies that, in the

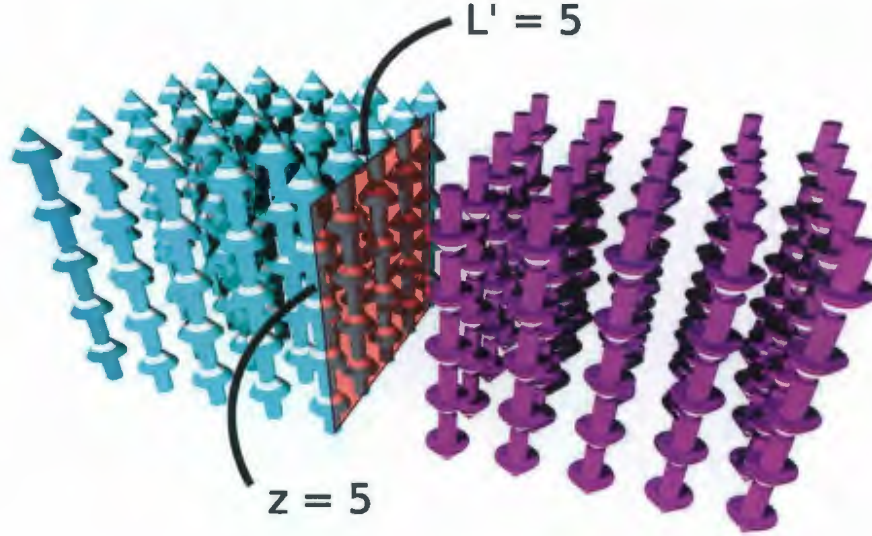


Figure 2.14: Two uniform Ising grains, each having $5 \times 5 \times 5$ spins with one having spins pointing up and the other having spins pointing down. The light red area on the left grain shows the spin that interact with the grain on the right. The area encompasses $L'z = 25$ spins such that here, $J_{\text{eff}} = 25J$.

case of a homogeneous multilayer film of thickness z , the transition temperature $T_c(z)$ is such that $T_c(1) = 2.269 < T_c(z) < T_c(\infty) = 4.512$. A model function to explain multi-layer effects has been proposed based on finite-size scaling arguments [30] given by

$$T_c(z) = T_c(\infty) \left[1 - \frac{b}{z^{1/\nu}} \left(1 - \frac{a}{z} \right) \right] = T_c(\infty) f(z). \quad (2.8)$$

Simulations determined that fairly accurate values of the parameters a , b and ν were: $a = 1.37572$, $b = 1.92629$ and the 3D Ising critical exponent $\nu = 0.6289$. This function is plotted in Fig. 2.16 and is considered accurate for $z > 3$.

While Eq. 2.8 is for a model with infinite spins in two of the dimensions, it allows a good comparison to simulations for multilayer models. It does not account for the

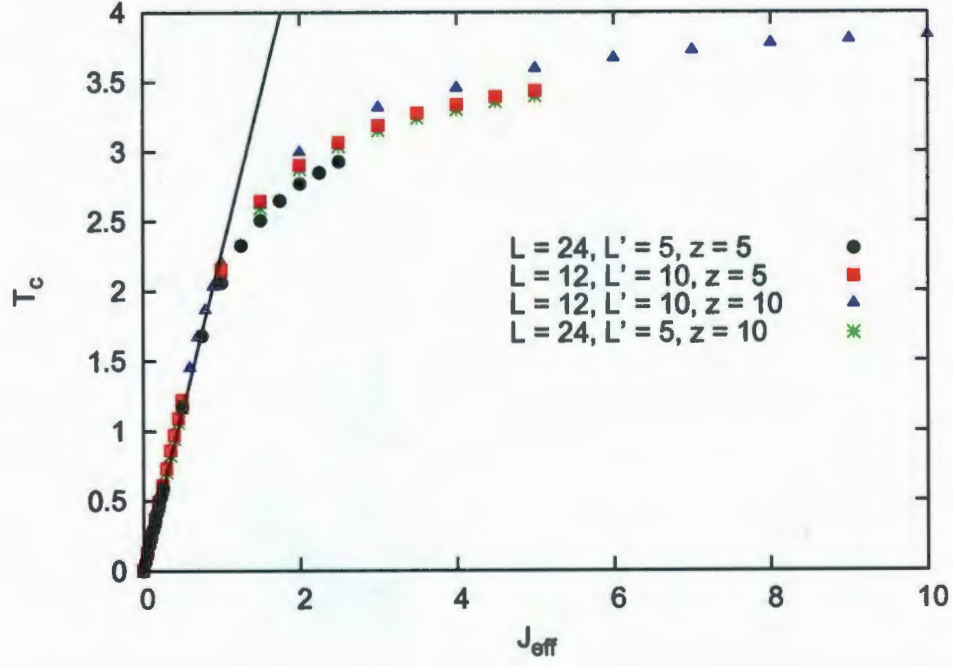


Figure 2.15: T_c vs. J_{eff} in the multi-layer case for different J , L' and z (values shown in the legend). Straight line has a slope of $2.269 \approx T_c/J_{\text{eff}}$ for the 2D model.

granular model, but modifications can be made to formulate a model to compare with simulation results.

To estimate $T'_c(L', z)$, it is noted that the correlation length for the 2D Ising model is given asymptotically by [31]

$$\xi'(T)^{-1} = 4\left(\frac{J'}{T} - \frac{J'}{T'_c}\right). \quad (2.9)$$

As the temperature nears the critical temperature, the correlation length diverges towards infinity; however, for the finite granular model, it cannot extend further than L' . As such, an estimation of T'_c can be made with the condition $\xi'(T'_c(L', z)) \approx L'$. Setting $T'_c = T_c(z)$ from Eq. 2.8, an expression is obtained for $T'_c(L', z)$

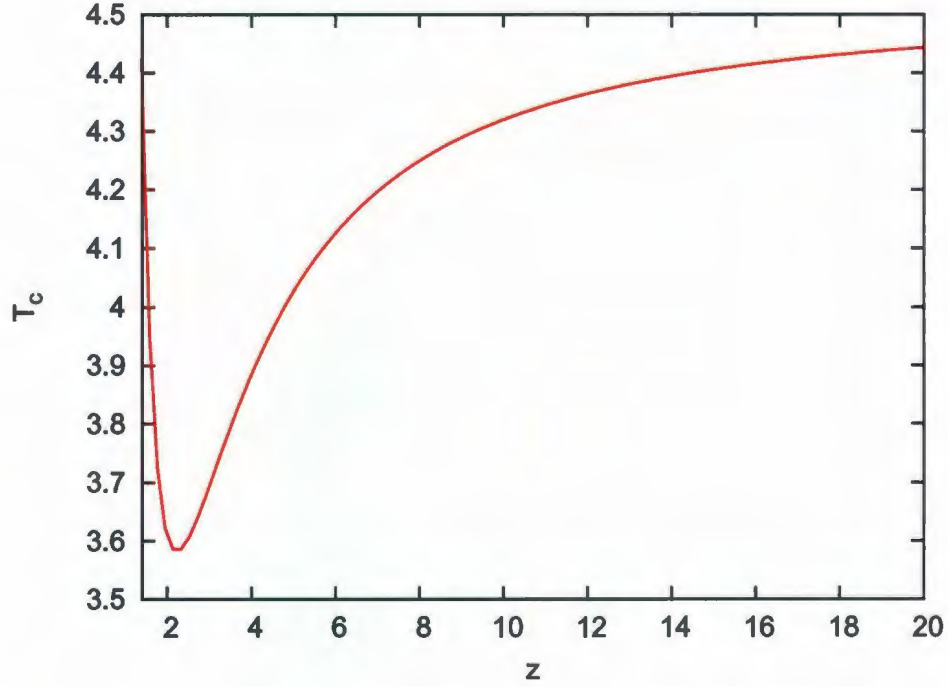


Figure 2.16: Plot of Eq. 2.8 showing the expected T_c as a function of z . The equation is only valid for $z > 3$.

$$T'_c(L', z) \approx \frac{T'_c(z)}{1 + T'_c(z)/4J'L'}. \quad (2.10)$$

Eq. 2.10 is an approximate extension of Eq. 2.8 for the granular model. Data obtained from applying it, along with simulation results, are included in Table 2.1 for $L' = 1, 5, 8$ and 10 . Despite the approximate nature of Eq. 2.10, comparisons show reasonable agreement with the simulation data.

This analysis suggests that for $T \ll T'_c$, the correlation length is very small compared to L' and the magnetization within a single grain can approximately be considered uniform and close to saturation. This suggests that the approximation of uniform magnetization within a grain is reasonable if $T_c \ll T'_c$.

Within the Ising model (at $H = 0$), a uniformly magnetized grain pointing either up

L'	5	8	10	$L' = \infty (J' = 1)$
$z = 1$	1.78(2.04)	(2.12)	2.04(2.15)	(2.269)
$z = 5$	3.15(3.35)	(3.57)	3.65(3.66)	$4.03 \pm 0.01(4.028)$
$z = 8$	(3.50)	3.7(3.75)	(3.84)	$4.25 \pm 0.05(4.246)$
$z = 10$	3.4(3.55)	(3.80)	3.9(3.90)	$4.32 \pm 0.01(4.317)$

Table 2.1: Intra-grain order temperature T'_c at which the peak in the heat capacity occurs as a function of the grain dimensions L' and z . The values in parentheses are calculated from the scaling relations from Eqs. 2.8 and 2.10.

or down contributes the same energy amount to the system Hamiltonian. This energy, E_0 , can be directly calculated as follows:

1. For a grain with $L'^2 z$ spins, each spin has six nearest neighbor interactions with energy $-J'$. However, since each interaction is counted only once, this is a total energy contribution of $-3J'L'^2 z$.
2. The spins on the top and bottom layer do not follow periodic boundary conditions and so an energy contribution of $-J'L'^2$ is not counted.
3. Similarly, as the calculation is done for the internal energy, the J interactions spins at the four grain edges are not counted. A total of $2 \times (-J'L'z)$ is to be taken out of the energy contribution.
4. The internal energy of a uniform grain is then $E_0 = -3J'L'^2 z - (-J'L'^2 - 2J'L'z) = -J'(3zL'^2 - L'(L' + 2z))$.

If every grain in the model is assumed to be uniformly magnetized, a single spin variable σ_I can be made to represent each one. These σ_I interact with neighbor grains with an interaction $J_{\text{eff}} = JL'z$ from Eq. 2.7. The granular Hamiltonian within the approximation is then given by

$$\mathcal{H}_{\text{grain}} = E_0 - J_{\text{eff}} \sum_{\langle II' \rangle}^{L^2} \sigma_I \sigma_{I'}. \quad (2.11)$$

This 2D model would have a critical temperature T_c given by $T_c \approx 2.269J_{\text{eff}}$. The deviation from this model is shown in Fig. 2.15.

It is possible to quantify, approximately, the region of validity for the uniform grain approximation using $T'_c(z, L) \gg T_c(z, L)$ as a criterion. Substituting Eq. 2.8 into Eq. 2.10

$$T'_c(L', z) \approx \frac{4.512J'f(z)}{1 + 4.512J'f(z)/4J'L'} \approx \frac{4.512J'f(z)}{1 + 1.128f(z)/L'} \approx 4.512J'f(z) \gg 2.269J_{\text{eff}}. \quad (2.12)$$

With $J' = 1$, $f(z) \approx 1$ and for sufficiently large L' , this reduces to $J_{\text{eff}} \ll 2.0$, which is consistent with the results plotted in Fig. 2.15 where the deviation happens at around $J_{\text{eff}} \approx 1.0$. For $z = 1$, $f(1) \approx 2.269/4.512 \approx 0.5$ and $J_{\text{eff}} = JL'z = JL'$, the inequality reduces to $J \ll 1.0/L'$ which is consistent with results plotted in Fig. 2.10. While this approach is very approximate, it does appear to be consistent with obtained simulation results.

Chapter 3

Ising Model With External Magnetic Field

3.1 M-H Loops: Background

Having a nonzero external magnetic field in the Ising model allows for the study of a slightly more complicated system, as the lattice loses its ground state symmetry. When $H = 0$ and $T = 0$, the energy of the system is the same if the spins are all pointing in the up direction or if they are all pointing down. As a consequence, the total system magnetization $M(T)$ of Eq. 1.23 was defined so as to take the absolute value of the sum of the spin magnetizations, to reflect this symmetry. When $H \neq 0$, the spins want to align themselves in the direction of H due to the Zeeman energy $-\mathbf{M} \cdot \mathbf{H}$, thus removing this degeneracy. In this case, it becomes important to differentiate between the spins pointing in one direction or the other. The total system magnetization when studying the Ising model with an external magnetic field is thus defined as

$$M(T) = \frac{1}{N} \left\langle \sum_i S_i \right\rangle. \quad (3.1)$$

By adding a magnetic field to the Ising model, it is also possible to study hysteresis

by varying H and observing the effect on M . A useful property that characterizes the degree of hysteresis is the coercivity (H_c), defined as the magnetic field at which the magnetization falls to zero. This is illustrated in Fig. 3.1, showing a hysteresis loop obtained on a homogeneous 2D Ising model.

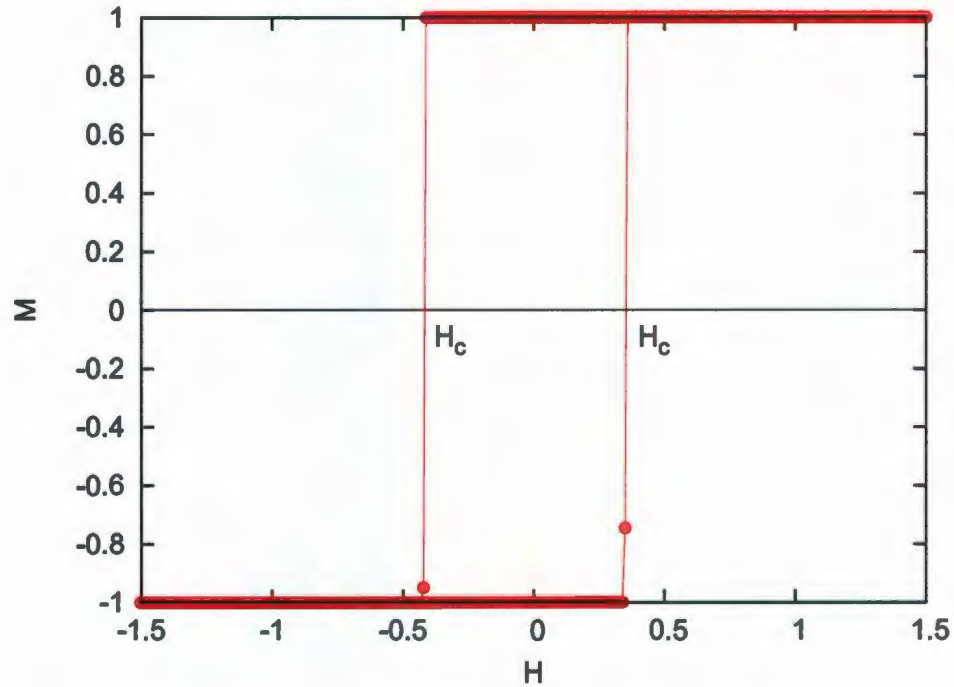


Figure 3.1: Hysteresis loop of the homogeneous 2D Ising model, showing the coercivity on both sides, where the magnetization sharply changes direction.

A material with high coercivity implies a robustness to superparamagnetic fluctuations, as a stronger magnetic field is needed to induce a flip in magnetization. For hard-drive recording media, this is essential so as to securely hold information in the magnetization of the bits. The coercivity depends on the temperature, falling to zero at the critical temperature. As HAMR relies on heating the material to high temperature, this implies a temporary lowering of the coercivity to be able to change the bit direction with an external field. To correctly model this process, it is essential to take into account the spin degrees of freedom internal to media grains, especially at temperatures close to

the critical temperature T'_c . This can be modeled using the granular Ising model, the principal focus of this thesis.

While it is certainly the case that Monte Carlo simulations show hysteresis below the critical temperature (as was shown in Fig. 3.1), its interpretation is not as straightforward, as hysteresis is a non-equilibrium, dynamical process. In the case of the classical Heisenberg model, it is possible to link the Metropolis algorithm to Langevin micromagnetics through the Fokker-Planck equation [32, 33], and hence relate the coercive field obtained from simulation results to experimental measurements. Unlike the classical Heisenberg model, the Ising model does not have any intrinsic dynamics associated with it [34], and thus simulation results shown in this chapter illustrate some useful trends regarding the impact of the inter-grain degrees of freedom on $H_c(T)$ and serve as an initial step towards a more realistic treatment based on the more general Heisenberg model.

3.2 Simulations

3.2.1 Homogeneous Ising model

Shown in Fig. 3.2 is a hysteresis loop of the homogeneous ($J = J'$) 2D single-layer Ising model for two different values of $J = J'$, 0.64 and 2.24 with $T/T_c \approx 0.21$ calculated with the Metropolis algorithm. Graphing M vs. H/J' , the data collapse onto a single curve.

This implies that the coercivity depends linearly on J' , as might be expected from simple scaling arguments. Fig. 3.3 shows the coercivity vs. temperature, illustrating the expected result that $H_c \rightarrow 0$ at T'_c . These results provide a baseline for simulations of the granular model.

3.2.2 Metropolis and Wolff Algorithms

In the case of the granular model, where there are large differences between the intra- and inter-grain exchange parameters J' and J , the equilibration times for $T < T'_c$ are

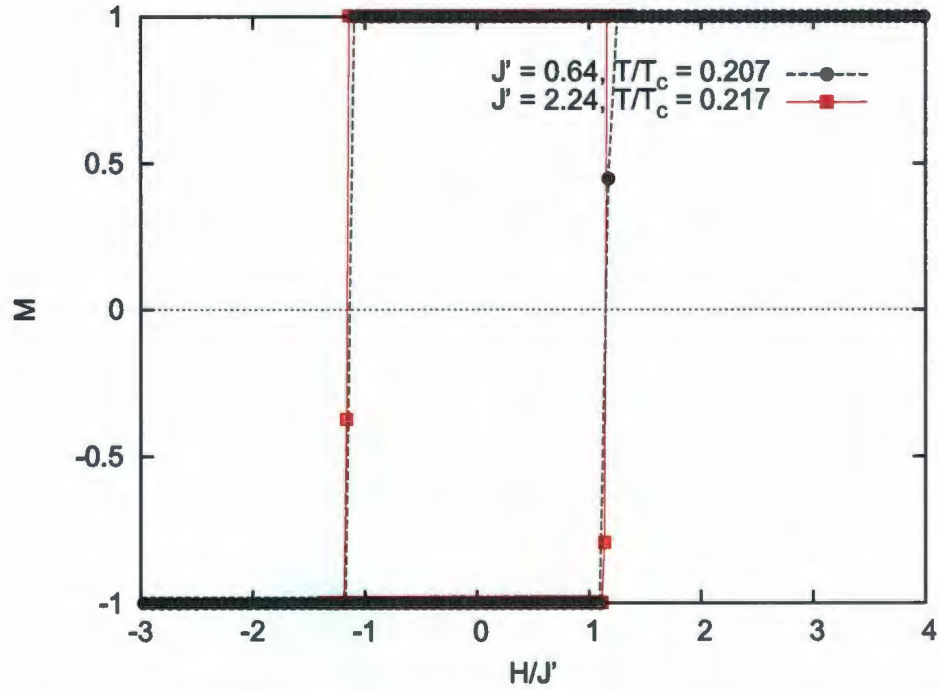


Figure 3.2: M-H loops for the homogeneous 2D Ising model using the Metropolis algorithm with M plotted as a function of H/J' for similar values of T/T_c .

strongly dependent on the choice of algorithm. In the case of the simple Metropolis algorithm, the acceptance rate for spin reversal drops off dramatically for $T \ll T'_c$. As a consequence, the number of Monte Carlo steps required to reverse a single grain increases significantly at low temperatures and, in order to reverse the magnetization, the simulation requires either large fields or many Monte Carlo steps. On the other hand, the Wolff algorithm, used to study the magnetization in the previous section, equilibrates in a reduced number of Monte Carlo steps by reversing large clusters. This yields a coercive field that is close to zero even down to the lowest temperature. These two situations are illustrated in Fig. 3.4 where Fig. 3.4(a) gives H_c vs. T when using only the Metropolis algorithm and Fig. 3.4(b) shows an M vs. H plot using the Metropolis combined with the Wolff algorithm.

There are a number of points worth noting from the coercivity field plotted in

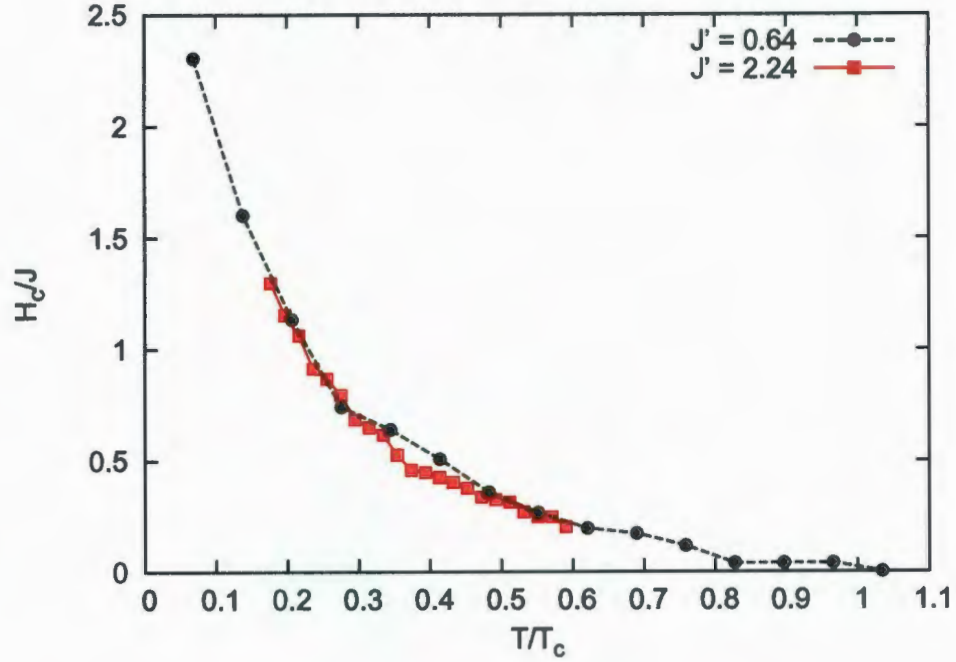


Figure 3.3: H_c/J vs. T/T_c of the homogeneous 2D Ising model calculated with the Metropolis algorithm for two values of $J = J'$ shown in the legend, giving the same profile.

Fig. 3.4(a). For example, the data show the coercivity dropping to zero at around $T = 3.7$, which corresponds to the T'_c of the system, as seen in Table 2.1. This is well above the value of T_c , where the total system magnetization goes to zero as would be expected. The data also show a systematic decrease in the coercivity with increasing MCS_T . Finally, it is noted that the shape of this $H_c(T)$ curve is very similar to corresponding results from micromagnetic simulations on an anisotropic Heisenberg model system of nanoparticles [35]. In contrast to the coercivity data shown in Fig. 3.4(a), the M-H loop obtained using the Wolff/Metropolis algorithm, shown in Fig. 3.4(b), exhibits no (or negligible) hysteresis.

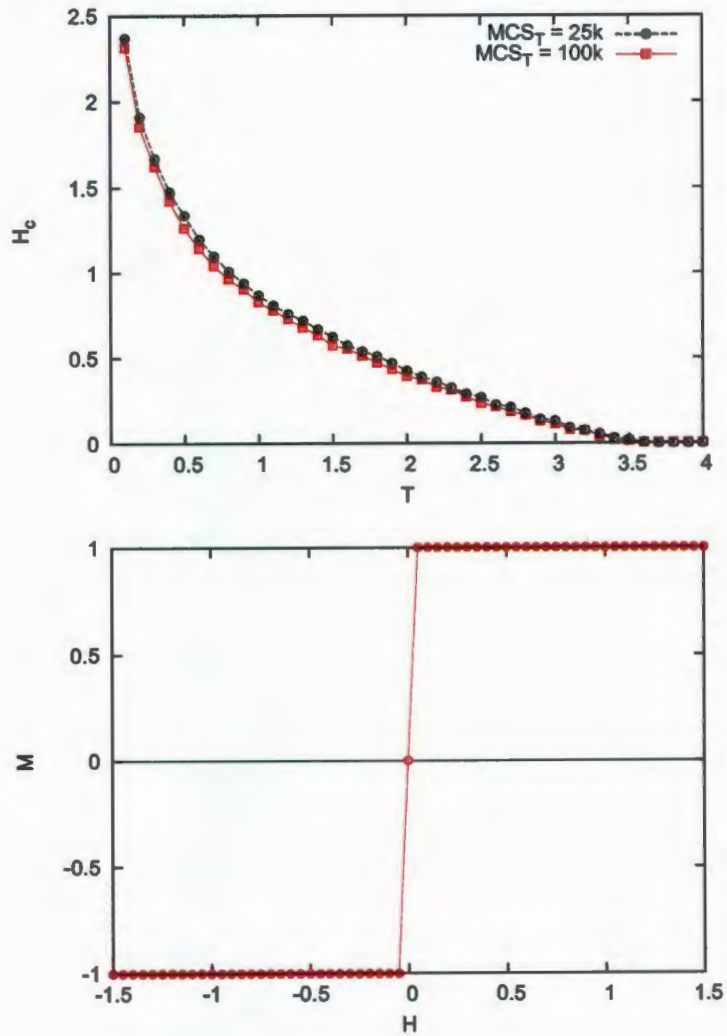


Figure 3.4: Fig. (a) shows H_c vs. T with only the Metropolis algorithm for the granular multilayer model, falling to zero at T'_c . Fig. (b) shows the first part of an M-H loop with the original Wolff/Metropolis algorithm for $T = 0.5$ having the same parameters as Fig. (a) with $MCS_T = 5000$. Here, $J' = 1$, $J = 0.01$, $L = 8$, $L' = 8$, $z = 8$ and MCS_T is shown in the legend for Fig. (a).

3.2.3 Wolff Grain Algorithm

The goal is then to develop a new algorithm specifically for this model which explicitly takes into account the grain flipping dynamics. A modified version of the Wolff algorithm

is developed, where a virtual cluster is not allowed to extend past a grain boundary which essentially means that a single application of this algorithm will never flip more than one grain. This Wolff grain algorithm constructs the virtual cluster in the same manner as the original Wolff algorithm, but sets $J = 0$ when considering spins at the edge, though only when constructing the cluster and not when calculating the energy. As single spins can still be considered, ergodicity is satisfied. The linking probabilities and the acceptance ratio are then determined so detailed balance is satisfied.

As the clusters are constructed in almost the same fashion, only the border spins of the virtual cluster will have an impact on $g(a, b)$ and $g(b, a)$, just like the original Wolff algorithm, and ΔE does not change. From the derivation of detailed balance for the Wolff algorithm with J' and J of Sec. 2.1, detailed balance is given by

$$\frac{P(a, b)}{P(b, a)} = \frac{g_0(1 - P_J)^m(1 - P_{J'})^{m'} A(a, b)}{g_0(1 - P_J)^n(1 - P_{J'})^{n'} A(b, a)} = e^{-\beta \Delta E}. \quad (3.2)$$

Since spins at the border of the grain are not considered, P_J is set to zero. Thus,

$$\begin{aligned} \frac{A(a, b)}{A(b, a)} &= (1 - P_{J'})^{-(m' - n')} e^{-\beta[2J(m - n)]} e^{-\beta[2J'(m' - n')]} e^{-2\beta k H S_t} \\ &= (e^{2\beta J})^{-(m - n)} [e^{2\beta J'} (1 - P_{J'})]^{-(m' - n')} e^{-2\beta k H S_t}. \end{aligned} \quad (3.3)$$

It can be seen that the dependence on m and n cannot be fully removed by a simple choice of $P_{J'}$. However, choosing again $P_{J'} = 1 - e^{-2\beta J'}$ does allow the dependence on m' and n' to be removed, so that

$$\frac{A(a, b)}{A(b, a)} = e^{-2\beta[J(m - n) + k H S_t]}. \quad (3.4)$$

This is more complicated due to m and n , which have to be tabulated. In the code, they are counted when the cluster is building as P_J (which is 0) is applied and the exponential can easily be calculated. The choice of $A(a, b)$ and $A(b, a)$ will depend on

whether $\{-2\beta[J(m-n) + kHS_i]\} > 0$ or not, with a positive exponent case causing $A(b, a)$ to be set to 1 and $A(a, b)$ to be $e^{-2\beta[J(m-n) + kHS_i]}$ and a negative exponent causing a switch in $A(b, a)$ and $A(a, b)$. To implement the acceptance ratio, m, n and then the exponential must be calculated in every case; if the exponential is greater than unity, the cluster is flipped. If it is smaller, generate a random number to compare with, much like the other algorithms. For similar reasons as the Wolff algorithm, this algorithm is used alongside the Metropolis algorithm, in the same ratio.

The motivation behind this modified Wolff/Metropolis algorithm involves the assumption that the spins within a single grain equilibrate much more rapidly than the grains themselves, which is particularly true for $T < T_c$ when the spins are mostly aligned in grains. The use of the Wolff algorithm to treat the spins within a grain ensure that the grains maintain a "quasi-equilibrium" as the grains themselves order in response to the applied magnetic field.

3.2.4 Granular Model Simulation results

For the model with no external field of Chap. 2, J_{eff} was defined to represent a grain face or area which allowed a data collapse for systems of different sizes. Upon adding an external field, an effective H can also be defined, H_{eff} . When looking at a single grain, J_{eff} corresponds to the interaction between its own magnetic moment \mathbf{M}_i and the \mathbf{M}_j magnetic moments of its neighbor grains. When concerned with the Zeeman term involving $-\mathbf{H} \cdot \mathbf{M}_i$, the energy dependency on the grain volume is considered. In an effective model where the grains are taken as a giant single spin, J_{eff} is the coupling constant between a grain moment and its neighbor while H_{eff} is the field that affects a single grain moment. As such, for grains composed of $L^2 z$ spins,

$$H_{\text{eff}} = HL^2 z. \quad (3.5)$$

Just like the graphs of Chap. 2 which were plotted vs. J_{eff} , the data here are plotted

as a function of H_{eff} instead of H . Using the modified algorithm, M-H loops were calculated for various values of J and lattice dimensions $L \times L' = 100$ and $z = L'$. The magnetization data are plotted in Fig. 3.5 as a function of the ratio $H_{\text{eff}}/J_{\text{eff}}$ using a range of parameters J and L' which give similar values of the ratio T/T_c . Data for the homogeneous model from Sec. 3.2.1 are also included in Fig. 3.5 for comparison.

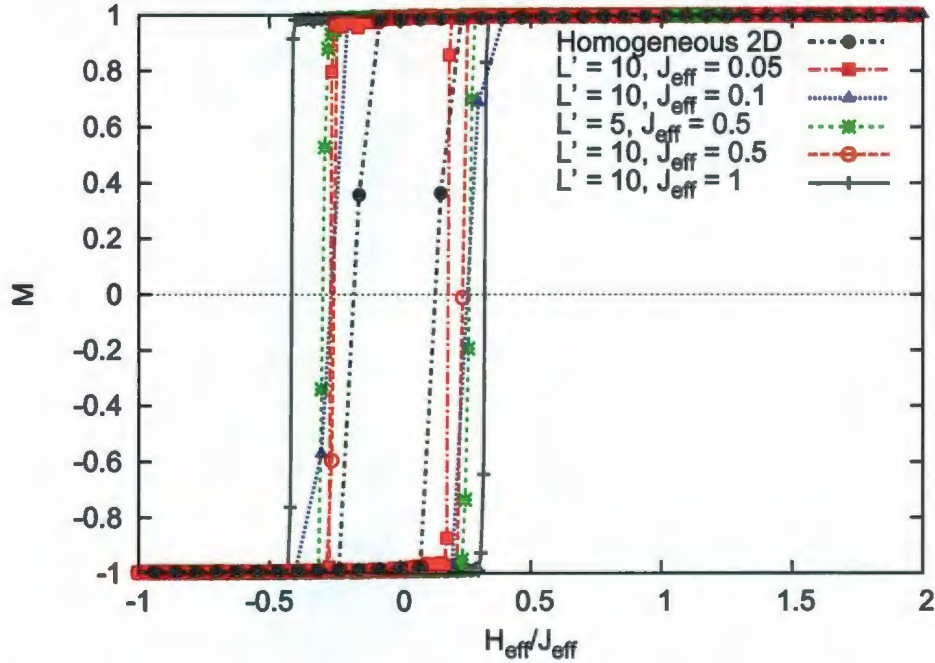


Figure 3.5: M-H loops calculated using the modified Wolff algorithm with the magnetization plotted as a function of $H_{\text{eff}}/J_{\text{eff}}$ for values of $T/T_c \approx 2/3$ using a range of J_{eff} and L' shown in the legend for the single-layer granular Ising model ($z = 1$). The data for the homogeneous 2D Ising model are also shown.

Roughly, a data collapse is expected for small J_{eff} , much like the granular model with no field, with a significant deviation for larger values of J_{eff} . A complete data collapse is not seen in Fig. 3.5, but the coercivity is in a reasonable range given the variation in J' , L' and z . There also does seem to be a general increase in H_c for higher values of J_{eff} .

To better illustrate the change in coercivity, Fig. 3.6 shows $H_{\text{eff}}^c/J_{\text{eff}}$ vs. T/T_c using the data in Fig. 3.5, where $H_{\text{eff}}^c = H_c L'^2 z$, as an analog to Eq. 3.5. The same effect of

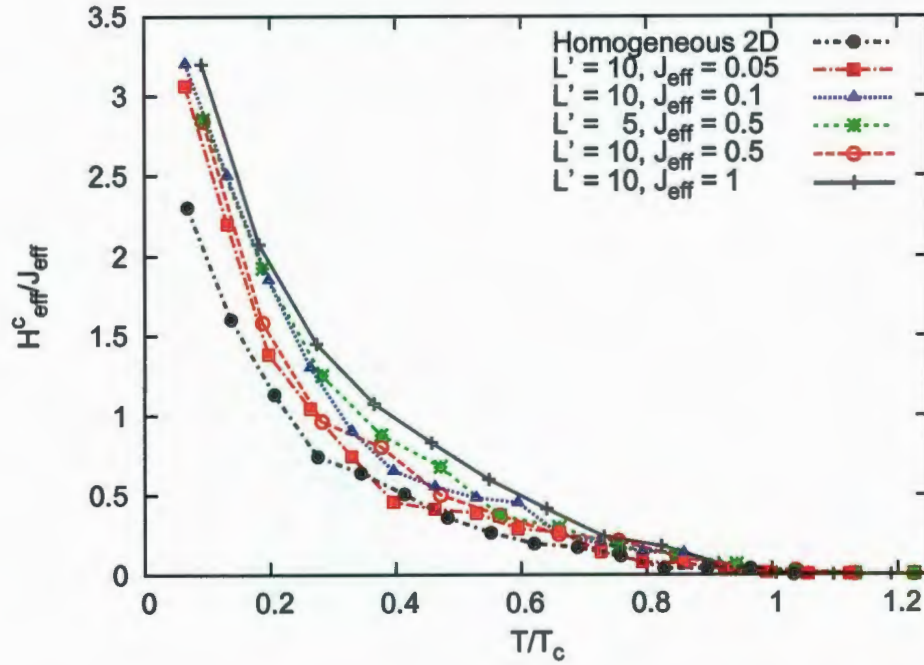


Figure 3.6: Plots of the scaled coercive field $H_{\text{eff}}^c / J_{\text{eff}}$ plotted as a function of T / T_c for several values of J_{eff} and L' shown in the legend for the single-layer granular Ising model ($z = 1$). The data for the homogeneous 2D Ising model are also shown.

stronger inter-grain coupling is seen here as previously at $H = 0$: a higher J_{eff} causes a stronger deviation (higher $H_{\text{eff}}^c / J_{\text{eff}}$) from the homogeneous 2D model data. For separate values of L' and J_{eff} , Monte Carlo simulations with the Wolff grain algorithm do provide reasonable qualitative estimates of the coercive field. The non-linear effect seen in the model with no external magnetic field isn't as clear to see here, but it does seem as though the quasi-dynamics implied by Monte Carlo simulations are affected by the internal spin degrees of freedom of the individual grains. To better understand these effects in real magnetic recording media, the more realistic Heisenberg model with anisotropy is studied in Chap. 4.

Chapter 4

Heisenberg Model

4.1 Heisenberg Model Simulation

While the Ising model does capture many of the essential features of highly anisotropic materials used in magnetic recording media, it cannot quantitatively be used to model dynamical and non-equilibrium properties. Dynamical effects of magnetic materials can be successfully simulated using the Heisenberg model. Unfortunately, the Heisenberg model is computationally more demanding than the Ising model and many of the methods used to simulate the Ising model are not readily generalised to the more general Heisenberg model.

Due to the additional computational demands associated with the Heisenberg model, simulations typically involve either a smaller MCS_T or a smaller system size. As the system size has a drastic effect on the run time of the program, smaller lattices are often used in the simulations while still trying to maintain a size big enough to capture accurate results.

4.1.1 Energy

The Heisenberg model described in Chap. 1 was given as

$$\mathcal{H} = - \sum_{\langle ij \rangle} J_{ij} \mathbf{S}_i \cdot \mathbf{S}_j - \sum_i \mathbf{H} \cdot \mathbf{S}_i - K \sum_i (S_i^z)^2. \quad (4.1)$$

Spins \mathbf{S}_i in the Heisenberg model are represented as three-dimensional unit vectors. The spin direction can be represented in spherical coordinates (r, θ, ϕ) with $r = 1$. In particular, since the spin vectors \mathbf{S}_i are not represented by discrete variables, the Boltzmann factor can no longer be simply described in terms of a simple and computationally efficient lookup table as described in Sec. 1.4.5, but instead requires a direct calculation.

4.1.2 Magnetization

For the Heisenberg model, the magnitude of the total system magnetization is defined by

$$M(T) = \frac{1}{N} \left\langle \sqrt{\left(\sum_N S^x \right)^2 + \left(\sum_N S^y \right)^2 + \left(\sum_N S^z \right)^2} \right\rangle, \quad (4.2)$$

while the magnitude of the grain magnetization, used for the granular model, is given by

$$M_g(T) = \frac{1}{N} \left\langle \sum_I \sqrt{\left(\sum_{N \in I} S^x \right)^2 + \left(\sum_{N \in I} S^y \right)^2 + \left(\sum_{N \in I} S^z \right)^2} \right\rangle, \quad (4.3)$$

where the I are the different grains on the lattice.

The direction that the spins point does not influence the order parameter, e.g., all the spins pointing in the \hat{z} axis or all the spins pointing in the \hat{y} axis will both give $M(T) = 1$. When the external magnetic field or the anisotropy is nonzero, the spins then have preferred direction.

4.1.3 Specific Heat and Susceptibility

The specific heat is defined the same way as the Ising model through energy fluctuations, though the calculation of the energy differs for the Heisenberg model. The susceptibil-

ity is a tensor with longitudinal and transverse components which can be calculated from spin fluctuations. For the present case, only the longitudinal component is considered which is defined the same way as the Ising model with the magnetization and magnetization squared defined as per Eq. 4.2.

4.2 Monte Carlo Algorithms for the Heisenberg Model

4.2.1 Metropolis

The Metropolis algorithm used in the Ising case can be generalized to the Heisenberg model with the spin flips, that define the Markov process in the case of the Ising model, replaced by a random rotation of the unit spins. Some care has to be taken to ensure that the random rotation generates a probability distribution that is uniform over the unit sphere. It can be shown [24] that this may be achieved by selecting the coordinates of the randomized spin such that $z = \cos \theta$ is distributed uniformly over the interval $(-1, 1)$ and ϕ is distributed uniformly over the interval $(0, \pi)$. To achieve this, θ itself is not uniformly generated, but comes from the distribution $\theta = \arccos(1 - 2R)$, where $0 \leq R \leq 1$ is uniformly generated.

The steps then for the Metropolis algorithm for the Heisenberg model can be stated as:

1. Pick a random spin on the lattice.
2. Calculate the energy of the system.
3. Generate two random numbers, $0 \leq R < 1$ and $0 \leq \phi' < 2\pi$.
4. Generate a new $\theta' = \arccos(1 - 2R)$
5. Calculate the energy if the spin \mathbf{S}_i is now defined by $r = 1$, θ' and ϕ' .
6. Calculate ΔE , the difference in energy between this energy and the previous one.

7. Calculate the Boltzmann weight, $w = e^{-\beta\Delta E}$.
8. Generate a third random number r such that $0 \leq r < 1$.
9. If $r < w$, flip the spin. If not, do nothing.

4.2.2 Wolff Algorithm

The changes required to adapt the Wolff algorithm to the Heisenberg model are somewhat more involved than those required to adapt the Metropolis algorithm described previously. The same basic idea is used, where a virtual cluster is constructed, but determining which spins can be added to the cluster is not as simple, as the binary choice of a cluster having spins pointing up or down cannot be applied. To get around this, an *Ising-like* variable is associated with the spin direction so that one can apply the same logic [26, 36]. Considering a single spin and the plane perpendicular to its spin direction, all the vectors that point in the same side of this plane have the same *Ising-like* spin variable $\delta = +1$. All the vectors pointing the other direction of the plane are considered to have $\delta = -1$. All the spins on the lattice can then be compared to a specific plane to check if they possess the same δ as the other spins.

The actual implementation of the Wolff Heisenberg cluster algorithm considers both a seed spin \mathbf{S}_s and the plane perpendicular to a seed unit vector $\hat{\mathbf{u}}_s$, which are both chosen randomly for every iteration of the Wolff algorithm. The easy way to determine which side of a plane a spin is pointing is with the scalar product, such that if $\mathbf{S}_s \cdot \hat{\mathbf{u}}_s > 0$, $\delta_s = +1$ and if $\mathbf{S}_s \cdot \hat{\mathbf{u}}_s < 0$, $\delta_s = -1$. When considering whether to add a spin \mathbf{S}_i or not to the virtual cluster, the first step is to compare δ_s with δ_i : if $\delta_s \neq \delta_i$, the spin \mathbf{S}_i is not be added to the cluster.

If, however, $\delta_s = \delta_i$, the spin is added with a certain probability, the linking probability. This probability is not the same as in the Ising model: to satisfy detailed balance, it becomes

$$P_J = 1 - \exp[-2\beta J(\mathbf{S}_i \cdot \hat{\mathbf{u}}_s)(\mathbf{S}_j \cdot \hat{\mathbf{u}}_s)], \quad (4.4)$$

where \mathbf{S}_i is the spin whose neighbor, \mathbf{S}_j , is the spin that might be added to the cluster. As spins are not added to the cluster if they do not share the same δ , the exponent will always be negative and P_J will thus vary between 0 and 1. The derivation of P_J is not as simple as the Ising case, but its verification is straightforward.

The spin $\mathbf{S}_i = S_i^x \hat{\mathbf{x}} + S_i^y \hat{\mathbf{y}} + S_i^z \hat{\mathbf{z}}$ can also be expressed as $\mathbf{S}_i = S_i^{u_s} \hat{\mathbf{u}}_s + S_i^{u'_s} \hat{\mathbf{u}}'_s + S_i^{u''_s} \hat{\mathbf{u}}''_s$ where $\hat{\mathbf{u}}'_s$ and $\hat{\mathbf{u}}''_s$ are unit vectors perpendicular to $\hat{\mathbf{u}}_s$, and $S_i^{u_s}$ can also be written as $\mathbf{S}_i \cdot \hat{\mathbf{u}}_s$. Upon reflection of the spin by the plane perpendicular to $\hat{\mathbf{u}}_s$, \mathbf{S}_i becomes $\mathbf{S}'_i = -S_i^{u_s} \hat{\mathbf{u}}_s + S_i^{u'_s} \hat{\mathbf{u}}'_s + S_i^{u''_s} \hat{\mathbf{u}}''_s$.

As the spin \mathbf{S}_j does not change upon the reversal of \mathbf{S}_i , ΔS_j is then $\Delta S_j = \mathbf{S}'_j - \mathbf{S}_j = 0$. For \mathbf{S}_i , it is found that $\Delta \mathbf{S}_i = \mathbf{S}'_i - \mathbf{S}_i = -2S_i^{u_s} \hat{\mathbf{u}}_s = -2(\mathbf{S}_i \cdot \hat{\mathbf{u}}_s) \hat{\mathbf{u}}_s$. This also explicitly shows the step needed to actually flip the spin, given by the reflection equation $\mathbf{S}'_i = \mathbf{S}_i - 2(\mathbf{S}_i \cdot \hat{\mathbf{u}}_s) \hat{\mathbf{u}}_s$. The energy change associated with nearest-neighbor interactions caused by this one spin's reversal, due to $E = -J \sum \mathbf{S}_i \cdot \mathbf{S}_j$ is given as, for one neighbor's contribution,

$$\begin{aligned} \Delta E_i &= -J(\mathbf{S}'_i \cdot \mathbf{S}_j - \mathbf{S}_i \cdot \mathbf{S}_j) \\ &= -J[(\mathbf{S}'_i - \mathbf{S}_i) \cdot \mathbf{S}_j] \\ &= -J[-2S_i^{u_s} \hat{\mathbf{u}}_s \cdot (S_j^{u_s} \hat{\mathbf{u}}_s + S_j^{u'_s} \hat{\mathbf{u}}'_s + S_j^{u''_s} \hat{\mathbf{u}}''_s)] \\ &= 2J(S_i^{u_s} S_j^{u_s}) \\ &= 2J(\mathbf{S}_i \cdot \hat{\mathbf{u}}_s)(\mathbf{S}_j \cdot \hat{\mathbf{u}}_s). \end{aligned} \quad (4.5)$$

From this, the factor $e^{-\beta \Delta E_i}$ becomes

$$e^{-\beta \Delta E_i} = \exp[-2\beta J(\mathbf{S}_i \cdot \hat{\mathbf{u}}_s)(\mathbf{S}_j \cdot \hat{\mathbf{u}}_s)]. \quad (4.6)$$

While this is only for one neighbor, consideration for the other nearest neighbors will only add an exponent factor, much like it did with $(m - n)$ in the Ising case, which cancelled out.

Once the virtual cluster is constructed, it is flipped with a certain probability, the acceptance ratio A . The definition of *flipped* here is, of course, different from the Ising model and different from the Heisenberg implementation of the Metropolis algorithm. In the same vein as the *Ising-like* spin variables, flipped spins merely undergo a reflection in the plane perpendicular to $\hat{\mathbf{u}}_s$. As $\hat{\mathbf{u}}_s$ is completely randomly generated at each iteration of the Wolff algorithm, the clusters built will often times be vastly different.

The acceptance ratio A will, just like the Wolff algorithm for the Ising model, depend on the energy due to the magnetic field. Furthermore, the energy associated with the anisotropy will also play a role, though its effect is very similar in nature to the magnetic field. Every spin that gets flipped changes the magnetic field and anisotropy contribution to the energy in a different way, so that to account for them, a running tally is kept.

For the Zeeman term, the effect of the cluster's reversal is

$$\Delta E_H = E'_H - E_H = \sum_i (\mathbf{H} \cdot \mathbf{S}'_i - \mathbf{H} \cdot \mathbf{S}_i) = \sum_i (-2S_i^{u_s} \mathbf{H} \cdot \hat{\mathbf{u}}_s). \quad (4.7)$$

For the anisotropy, a simple expression such as the one for the magnetic field cannot be found, but rather is expressed as

$$\Delta E_K = E'_K - E_K = \sum_i [K(S_i'^z)^2 - K(S_i^z)^2] = \sum_i K[(S_i'^z)^2 - (S_i^z)^2]. \quad (4.8)$$

Due to the squared terms, this cannot be simplified; however, it can be computed directly by calculating $S_i'^2 - S_i^2$ for every spin in the cluster and then adding them up.

The acceptance ratio is then $A = e^{-\beta(\Delta E_H + \Delta E_K)}$ which becomes

$$A = \exp\{\beta[2(\mathbf{H} \cdot \hat{\mathbf{u}}_s)(\sum_i \mathbf{S}_i \cdot \hat{\mathbf{u}}_s)] - [K \sum_i (S_i'^z)^2 - (S_i^z)^2]\}. \quad (4.9)$$

The steps needed to apply the Wolff algorithm for the Heisenberg model can then be summarized as:

1. Select a random seed spin \mathbf{S}_s and generate a random unit vector $\hat{\mathbf{u}}_s$.
2. Calculate the reflection of the spin from the plane that is perpendicular to $\hat{\mathbf{u}}_s$.
3. Calculate $\delta_s = \text{sgn}(\mathbf{S}_s \cdot \hat{\mathbf{u}}_s)$.
4. Look at all the nearest neighbors of the seed spin and for each one, calculate their δ_j in the same way.
5. If $\delta_s = \delta_j$, add the spin to the cluster with probability P_j .
6. For each spin added to the cluster, look at their nearest neighbors and follow the same procedure until there are no more spins considered. There can be multiple opportunities for spins to be added to the cluster due to the way the cluster is constructed.
7. Generate a random number $0 \leq r < 1$ and calculate the acceptance ratio A .
8. If $r < A$, reflect every spin \mathbf{S}_i in the cluster using $\mathbf{S}'_i = \mathbf{S}_i - 2(\mathbf{S}_i \cdot \hat{\mathbf{u}}_s)\hat{\mathbf{u}}_s$.

When applying the Wolff algorithm to the granular model, the only change is again with $P_{j'}$ and P_j . This proves to be quite efficient at helping the system equilibrate.

4.3 Heisenberg Model Simulation Results

One important distinction between the Ising model and the Heisenberg is that, unlike the Ising model, the 2D model with $H = 0$ and $K = 0$ does not exhibit long-range magnetic order for $T > 0$. In order to study the effects of granularity in the Heisenberg model, the 3D case and the 2D case with finite perpendicular anisotropy are considered here. Out of convention, $\hat{\mathbf{K}} = \hat{\mathbf{z}}$ defines the perpendicular anisotropy.

The ultimate goal for this study in examining the Heisenberg model concerns thin films, with a multilayer model. For comparative purposes, simulations were done on the 3D model, as it does have a phase transition for non-zero temperature at $K = 0$. Simulations were then done on the granular multilayer Heisenberg model, with results from the single-layer case for comparison.

4.3.1 3D Heisenberg Model

Homogeneous Model

Fig. 4.1 shows an example of the magnetization for different values of J in the homogeneous 3D model with $K = 0$. Here, $L = 10$ and $\text{MCS}_T = 10000$ using the Metropolis and Wolff algorithms.

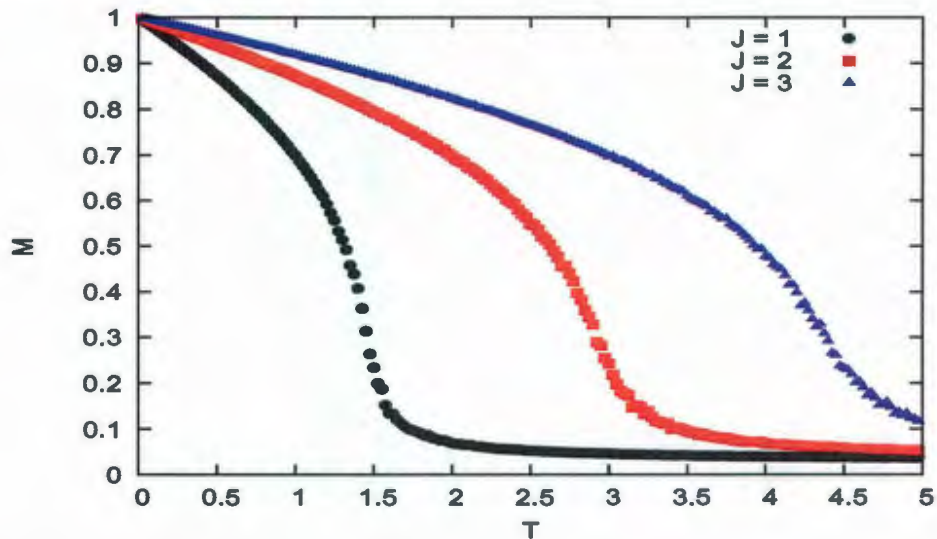


Figure 4.1: Magnetization vs. temperature for different values of $J = J'$ shown in the legend for the homogeneous 3D Heisenberg model with $K = 0$.

Despite the small system size and small number of Monte Carlo steps, clear evidence of the phase transition can be seen in the magnetization curves. Compared to the Ising model, the total magnetization below T_c is lower and remains well below 1 over

a larger range of temperatures. By looking at Fig. 4.1, T_c can be approximated to be slightly below $1.5J$ by looking at where the magnetization extrapolates to zero, which is consistent with the most accurate calculations of $T_c \approx 1.443J$ for the isotropic 3D classical homogeneous Heisenberg model [36].

Granular Model

While the results presented in Fig. 4.1 show that the implementation of both the Metropolis and Wolff algorithms successfully reproduce the results for the homogeneous model, as indicated in the introduction, magnetic recording media are highly anisotropic. The anisotropy energy of a grain scales linearly with its volume, $E_K \propto KV_g$, and it has been estimated that K values in the range $0.001 < K < 0.1$ (assuming $J' = 1$) are relevant to real media. In the present work, larger values are also considered for completeness. It is anticipated that, as was the case for J and H , a K_{eff} can be defined to describe the anisotropy's effect on grains, as will be shown below.

To better visualize the granular Heisenberg model, Fig. 4.2 shows an $80 \times 80 \times 80$ spin ($L' = 5$, $L = 16$) cube with the magnetization given by arrows drawn from a colormap. This particular picture was taken at the end of a $\text{MCS}_T = 100000$ run. It shows that while the individual spins within the grains are ordered, the grain themselves are not.

The magnetization in the Heisenberg model with grains ($J = 0.01$) and strong anisotropy ($K = 10$) is shown in Fig. 4.3. A small system size ($L = 2$, $L' = 5$) is used as these simulations were done with the 3D lattice.

It is seen that the system exhibits a large fluctuation in the total magnetization M for $T < 2.2$. However, these large fluctuations are not present in the grain magnetization M_g , which suggests that the spins inside the grains align themselves as expected, while the weakly-coupled grains do not exhibit long-range magnetic order.

Fig. 4.4 shows the effect on the magnetization caused by changing the anisotropy by several orders of magnitude. A lower anisotropy reduces the effect of the fluctuations as well as the transition temperature. The larger fluctuations at high K are a result

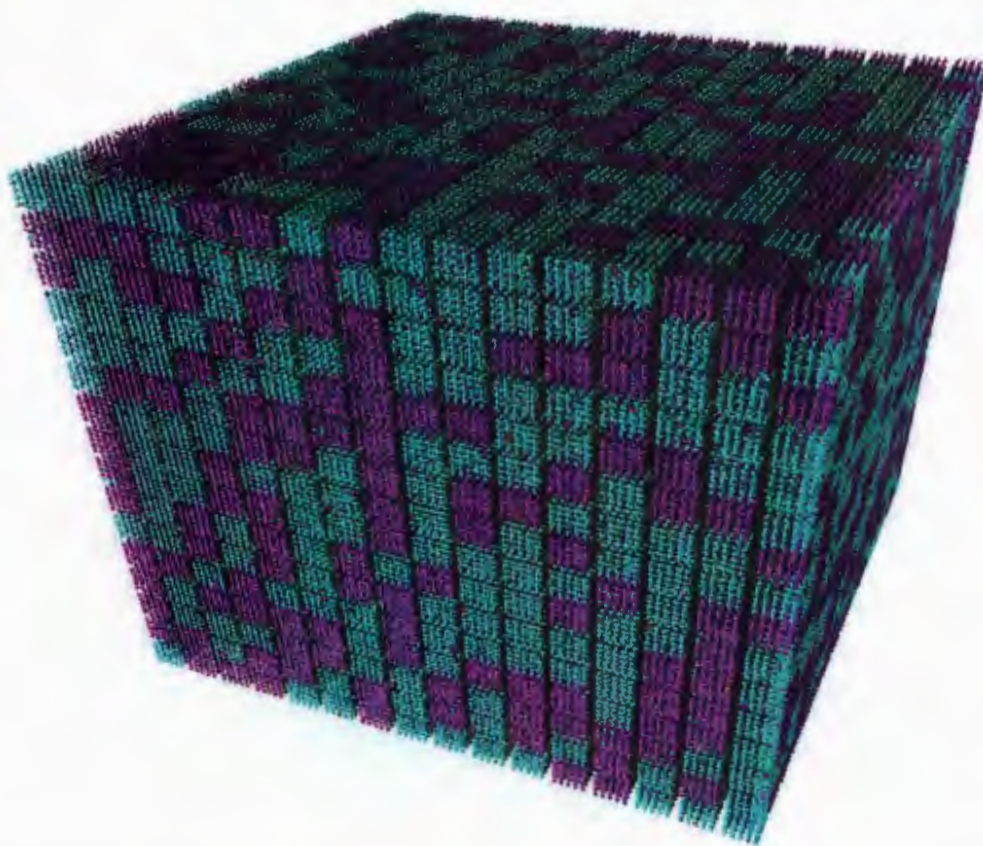


Figure 4.2: Granular 80^3 spin Heisenberg cube showing the magnetization with the positive z pointing perpendicular to the top face of the cube. Here, $L' = 5$ and $L = 16$. A purple color represents a negative z spin direction while a blue-green color is the positive z spin direction.

of a reduction in the Monte Carlo acceptance probability due to an increase in uniaxial anisotropy which tries to keep spins pointing along one axis.

Fluctuations

To study the effects of thermal fluctuations, extended runs of the magnetization vs. MCS_T for larger systems were performed at different values of K at specific temperatures. This is done to determine the values of MCS_T , system size and K that can be used in order to obtain reliable results. Fig. 4.5 shows the equilibration when $K = 10$, which

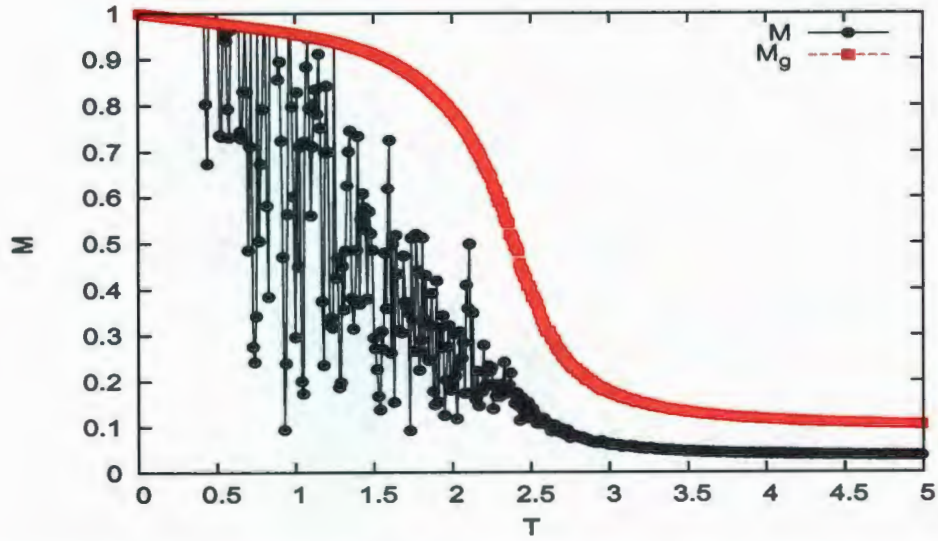


Figure 4.3: Total magnetization (M) and grain magnetization (M_g) vs. temperature for the 3D granular Heisenberg model. Here, $L = 2$, $L' = 5$, $MCS_T = 25000$, $J = 0.01$ and $K = 10$.

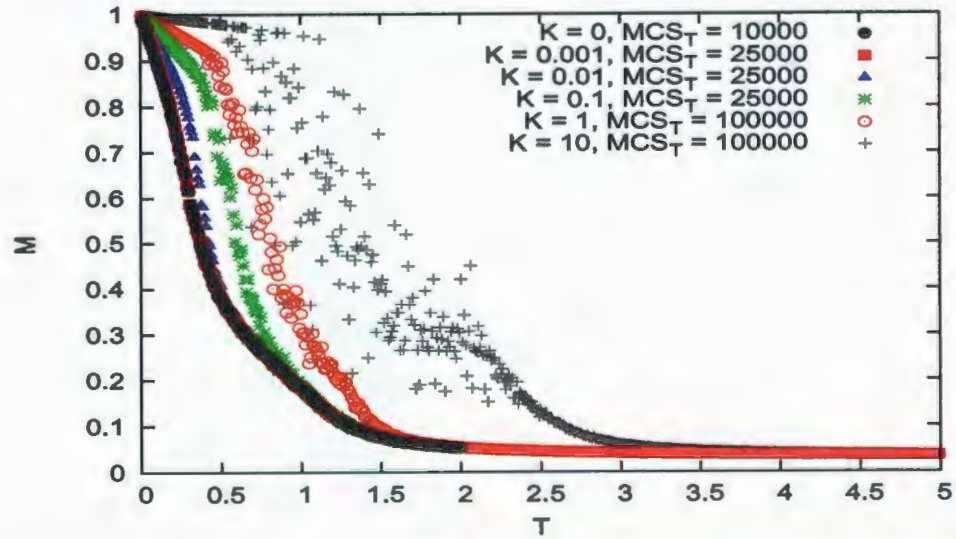


Figure 4.4: Total magnetization vs. temperature for different values of the anisotropy shown in the legend. Here, $L' = 5$, $L = 2$ and $J = 0.01$ for the 3D model.

has big fluctuations, also seen in Figs. 4.3 and 4.4.

The temperatures were chosen to be around or below T_c (which seems to be located

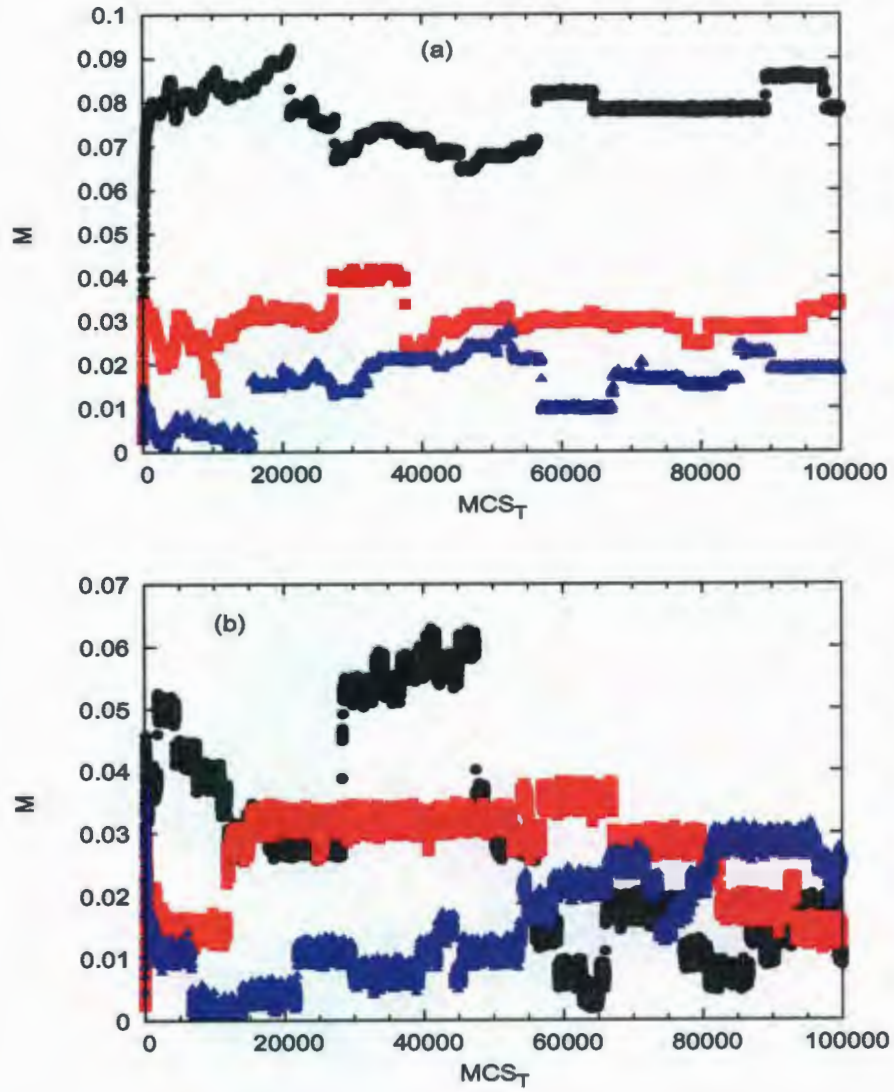


Figure 4.5: Magnetization vs. MCS for Fig. (a) $T = 1.5$ and Fig. (b) $T = 1.0$. Here, $L' = 5$, $L = 8$, $J' = 0.01$ and $K = 10$ for the 3D model. The different data series represent independent runs.

between $T = 1.5$ and $T = 2$), where the grains should begin to correlate and the fluctuations are largest. For both temperatures, three different runs are shown in Fig. 4.5, which are expected to give the same average magnetization after discarding MCS_0 steps.

What is seen is the magnetization reaching a value near its average very quickly (less than 1000 steps), but the average magnetization is not the same for the three runs. Even though these runs are done for a large $MCS_T = 100000$, the magnetization does not stabilize. Discrete steps are seen where the magnetization hovers around a specific value for several thousand steps before jumping to another value. As this is around or below T_c , these correspond to entire grains flipping.

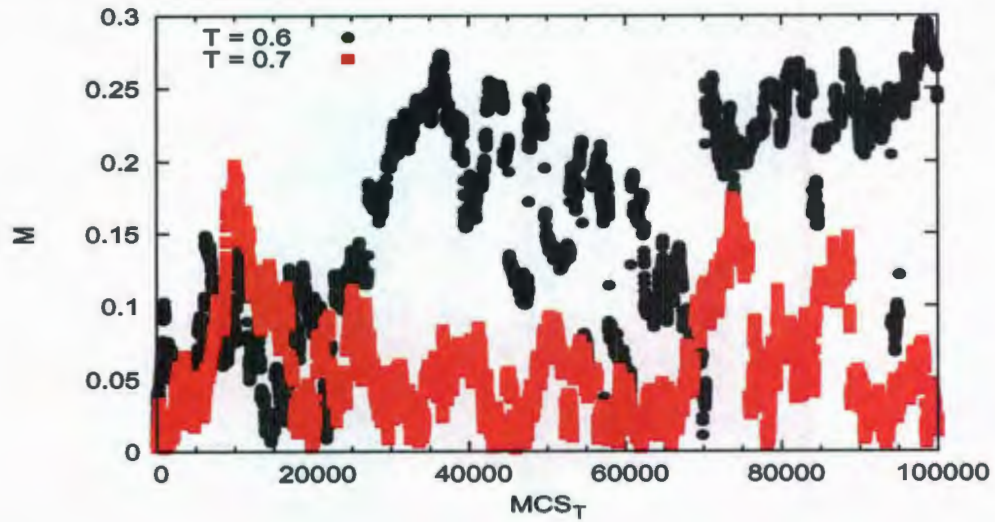


Figure 4.6: Magnetization vs. MCS at $T = 0.6$ and $T = 0.7$ with $K = 0.1$. Here, $L' = 5$, $L = 8$ and $J = 0.01$ for the 3D model.

Fluctuations are also examined with different anisotropy values in Figs. 4.6-4.8. Fig. 4.6 shows results with $K = 0.1$ and shows sizable fluctuations, where the magnetization does not stabilize much like the effect also seen at $K = 10$ in Fig. 4.5. Fig. 4.7 shows results at $K = 0.01$ with $T = 0.4$ and illustrates also the finite size effect of L . $L = 2$ and $L = 4$ show big variations in the magnetization, but have around the same average value. The $L = 8$ case shows a magnetization that is much smoother. The magnetization is more stable than in the other data seen so far, although big dips are still seen. A higher MCS_T is needed to get an accurate average value. Lastly, Fig. 4.8 shows the magnetization below and around the critical temperature. As seen in Fig. 4.4,

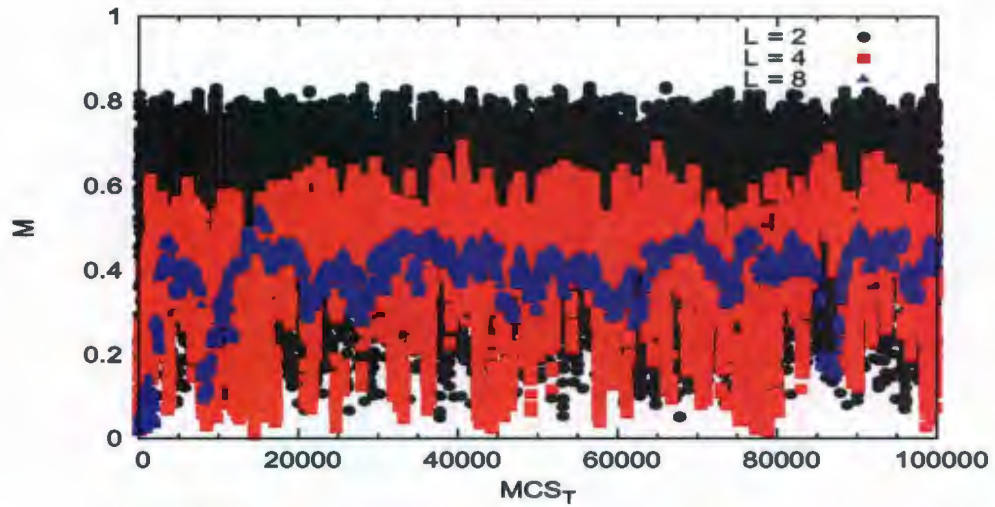


Figure 4.7: Magnetization vs. MCS for different values of the number of grains L for $K = 0.01$. Here, $L' = 5$, $T = 0.4$ and $J = 0.01$ for the 3D model. L is shown in the legend.

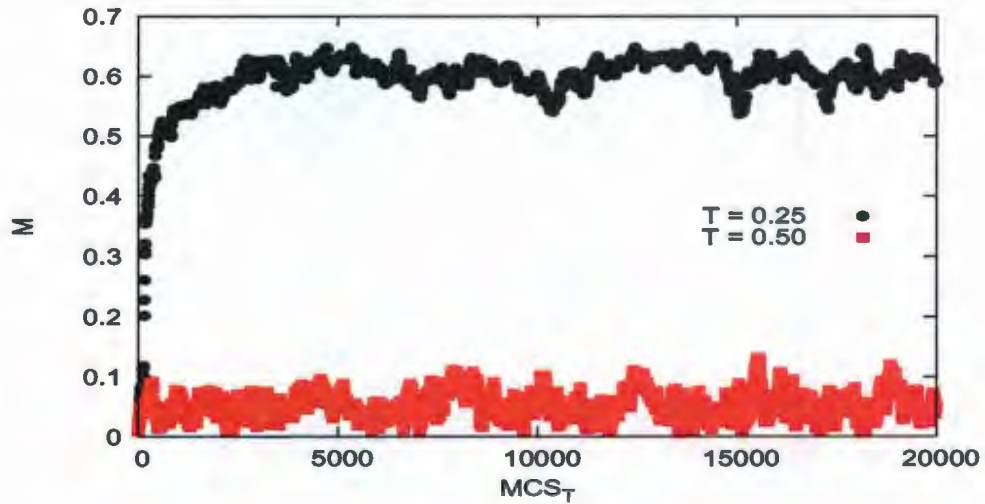


Figure 4.8: Magnetization vs. MCS for different values of the temperature with $K = 0.001$. Here, $L' = 5$, $L = 8$ and $J = 0.01$ for the 3D model. T is shown in the legend.

the data with $K = 0.001$ does not exhibit strong fluctuations. After some hundreds of steps, the magnetization hovers around its average value without much deviation.

Specific Heat and Susceptibility

The effect of K on the specific heat and the susceptibility was also studied. As the susceptibility is calculated from the magnetization, the fluctuations seen with M are seen in the susceptibility, especially for higher values of K . Fig. 4.9 shows both C and χ for various values of the anisotropy.

The susceptibility is very similar to the Ising model, with its single peak at T_c . As can be seen, larger fluctuations occur at larger values of K , especially near T_c . The zero-temperature specific heat is different from the Ising case with a value around 1 instead of 0. The value of the specific heat is not very stable for specific temperatures, although this is mostly due to the modest value of MCS_T used. This makes the determination of T'_c more difficult compared to the Ising case. It is seen that changing the anisotropy does not affect C at the lower values of K , but for $K = 1$, T'_c can increase significantly.

4.3.2 Thin Film Heisenberg Model

Homogeneous 2D Model

While the 2D Heisenberg model exhibits no phase transition when there's no anisotropy, it is possible to extract a T_c for $K \neq 0$ from the behavior of the susceptibility. Fig. 4.10 shows T_c vs. K taken for two different values of $J = J'$ (no grains).

As expected, T_c goes to zero as K approaches zero. Plotting the data as T_c/J vs. K/J , the two datasets collapse into a single curve as may be expected based on simple scaling arguments. It is expected that the results from the multilayer ($z > 1$) granular Heisenberg model collapse onto the data from this graph, at least for small J_{eff} , much like the Ising case.

Fig. 4.11 shows T_c vs. J , for specific values of K in the homogeneous 2D Heisenberg model. A linear behavior can be observed at the lower anisotropy values. For higher ($K = 5, 10$) anisotropies, a non-linear deviation is seen. The goal is then to take Figs. 4.10-4.11 which provide a homogeneous model viewpoint and collapse data from the

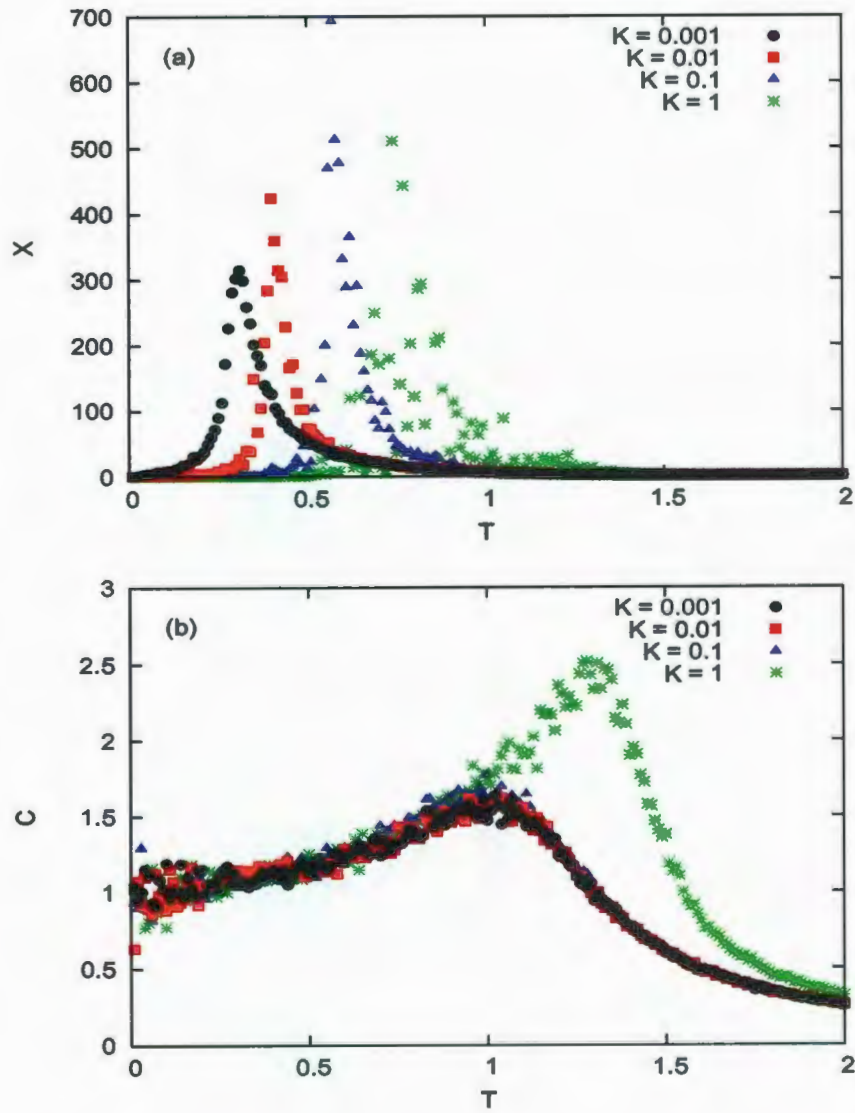


Figure 4.9: Fig. (a) shows the susceptibility while Fig. (b) shows the specific heat of the granular 3D Heisenberg model for values of K shown in the legend. Here, $L = 4$, $L' = 5$, $MCS_T = 25000$ and $J' = 0.01$.

granular model onto it as was done in the Ising case with the use of effective parameters, J_{eff} and K_{eff} .

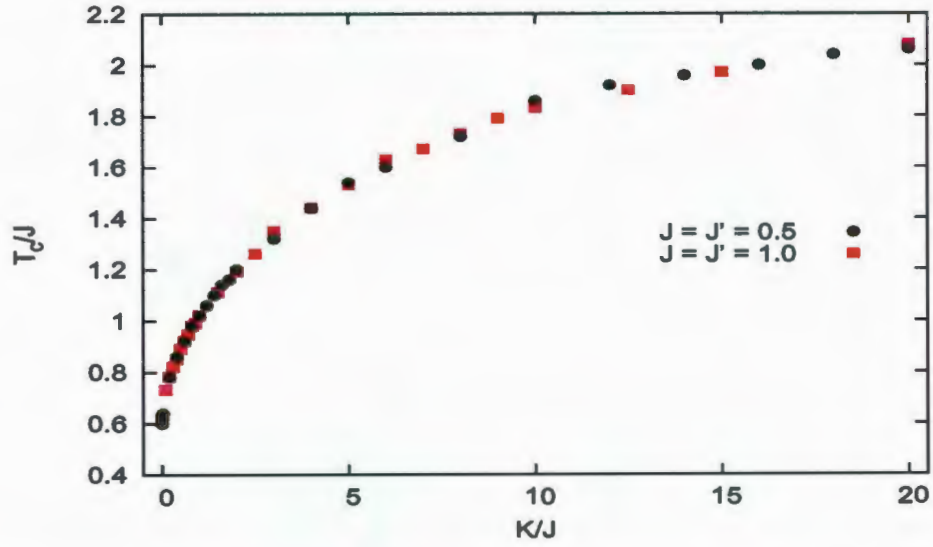


Figure 4.10: T_c/J vs. K/J of the 2D homogeneous Heisenberg model for $J' = J = 1$ with $L = 80$ and $J' = J = 0.5$ with $L = 160$.

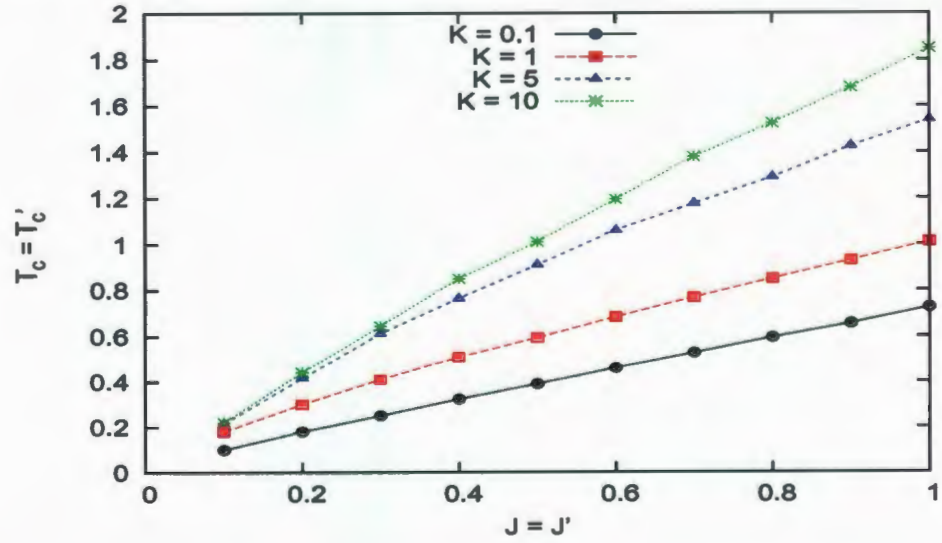


Figure 4.11: T_c vs. J of the 2D homogeneous Heisenberg model for different values of K (shown in the legend). Here, $L = 80$.

Multilayer Granular Model

Using the same energy scaling arguments that lead to the definition of H_{eff} , it is possible to define here

$$K_{\text{eff}} \equiv KL'^2z. \quad (4.10)$$

The results presented used $L' = z = 5$, so that $K_{\text{eff}} = 125K$, a difference of two orders of magnitude, which is significant. As big fluctuations are present at large values of the anisotropies, the values of K that can be used with moderate simulation times are thus fairly limited.

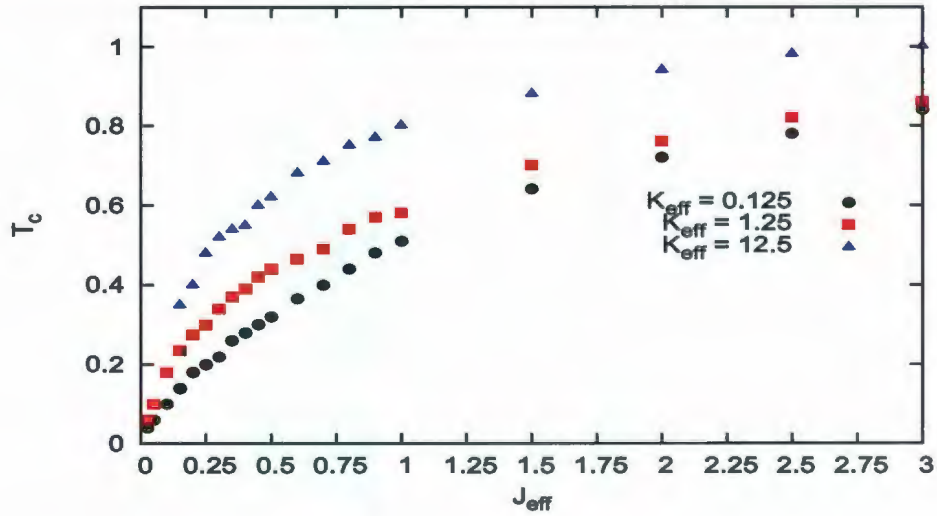


Figure 4.12: T_c vs. J_{eff} with different values of K_{eff} shown in the legend for the multilayer granular Heisenberg model.

Fig. 4.12 shows the result of T_c vs. J_{eff} ($J_{\text{eff}} = JL'z$) for different values of K_{eff} . Unlike what was seen in Fig. 4.11, the results here do not appear to show linearity. However, having $K_{\text{eff}} > 1$ deviates from the estimates of $0.001 < K < 0.1$ for real media and such cases are thus not as important. Taking these data for the granular model and collapsing it onto the homogeneous model is then the next goal. Fig. 4.13 shows a step in that direction.

Here, the K_{eff} shown in the legend in the homogeneous model cases are chosen to be the same as the ones for the granular model. For low values of J_{eff} , T_c for the

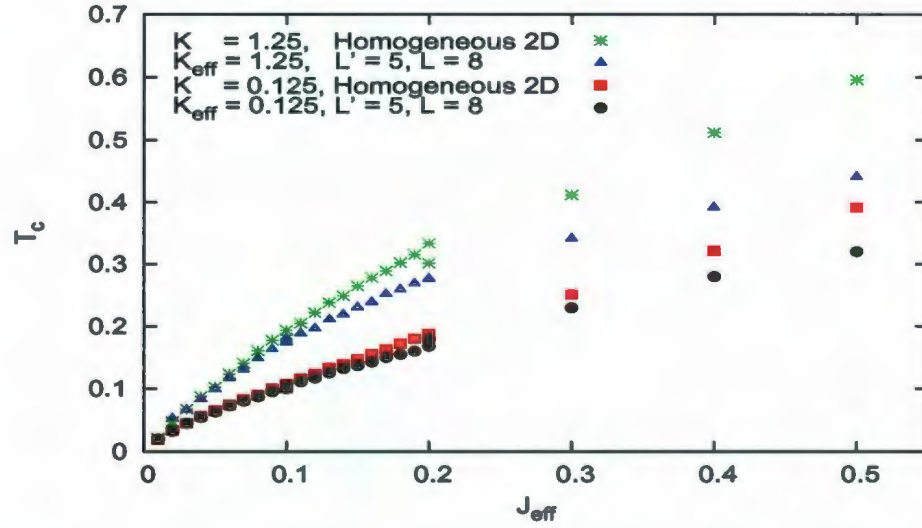


Figure 4.13: T_c vs. J_{eff} with two different values of K/K_{eff} shown in the legend comparing the multilayer granular and 2D homogeneous Heisenberg models.

homogeneous and granular models coincide before deviating at $J_{\text{eff}} \gtrsim 0.1$ for $K_{\text{eff}} = 0.125$ and $J_{\text{eff}} \gtrsim 0.05$ for $K_{\text{eff}} = 1.25$. This suggests a behavior similar to what was seen with the Ising case, where a grain can have a uniform magnetization below a certain J_{eff} . The value at which this deviation occurs then depends on K_{eff} , adding another complexity to the model. As such, the agreement of the homogeneous and granular model using effective parameters for the Heisenberg case may be less straightforward; however, it does appear to preliminarily indicate a regime of agreement which merits further studies.

Chapter 5

Discussions and Conclusions

A number of results have been presented which were obtained from a series of Monte Carlo simulations of the Ising and Heisenberg models. The granular model allows a study of the interplay between the fluctuations within the grains via exchange coupling J' and the fluctuations associated with weaker inter-grain coupling J . Various thermodynamic quantities such as the total magnetization M , grain magnetization M_g , specific heat C and susceptibility χ were used to probe the details concerning intra- and inter-grain order. Estimates of the temperature at which the two orderings occur, T'_c and T_c , were obtained as a function of the effective inter-grain exchange. The simulations used the Wolff algorithm to address the difficulties associated with equilibrating a system having two distinct energy scales. Preliminary results on hysteresis M - H loops based on the Ising model were also presented.

5.1 Ising Model

The work of Chaps. 2 and 3 present a detailed study of the granular Ising model, without and with an external magnetic field, to simulate magnetic recording media and field-induced switching. It is noted that the Ising model provides only an approximate representation of recording media which is better modeled by the Heisenberg Hamilto-

nian with strong anisotropy.

The use of the Wolff algorithm for the granular Ising model helped tremendously in obtaining accurate results without the use of a prohibitive amount of Metropolis Monte Carlo steps. Without this algorithm, the simple Metropolis method could not produce signature results of the model, such as two separate peaks in the specific heat and an accurate measure of T_c . The Wolff algorithm, taking into account the grain boundaries, is well suited to the granular model, as it incorporates the correct temperature dependent length scale.

A principal result coming from the analysis of the Ising model is found in Chap. 2 where a rough estimate of T'_c was found to be given by Eq. 2.12

$$\frac{T'_c(L', z)}{T_0} = \frac{f(z)}{1 + 1.128f(z)/L'}, \quad (5.1)$$

with $T_0 \approx 4.512J'$ being the Curie temperature for the 3D model and $f(z)$ is defined in Eq. 2.8. This result provides a quantitative estimate of an important part of the granular model based simply on system parameters.

It was also found that the inter-grain transition temperature T_c , estimated from the peak in the susceptibility, was consistent with the expected linear result $T_c \approx 2.27J_{\text{eff}}$ for sufficiently low $J_{\text{eff}} = JL'/z$. Within this regime, the temperatures at which the inter and intra-grain ordering occur are well separated with $T_c \ll T'_c$ and the fluctuations within the individual grains do not play a significant role in the inter-grain ordering. This already suggests that the assumption of uniform grain cannot be used in the HAMR approach; however, simulations using the more realistic Heisenberg model are needed to get better quantitative predictions.

Simulated M-H loops for the single-layer granular Ising model done to examine the role of the intra-grain fluctuations on the coercive field also provided interesting results. The modified Wolff algorithm allowed the capture of phenomena on a time scale associated with grain reversal without foregoing the intra-grain details. The coercivity

obtained with the modified Wolff algorithm goes to zero at $T/T_c = 1$, much like the magnetization on simulations without an external field. Using only the Metropolis algorithm, H_c does not go to zero until T'_c , which does not accurately represent its true behavior. As such, the modified Wolff algorithm does provide data which follows an expected trend. The results are at best qualitative, as the Ising model does not possess intrinsic dynamics like the Heisenberg model. This, however, does not prevent simulations from providing useful information, as was seen from Chap. 3. A systematic deviation is seen with increasing J_{eff} , which is consistent with the results from the simulations done without a magnetic field. Data from lower values of J_{eff} do not directly collapse on the homogeneous model results, but they do fall below the data from the higher values of J_{eff} as expected.

5.2 Heisenberg Model

The Heisenberg model presents a more realistic representation of recording media but also a more complicated model if only for the addition of the extra anisotropy parameter. For computational time reasons, the analysis that was done in this case was with smaller values of L , L' and MCS_T , which gives rise to larger finite size effects. Interesting results were still found for this anisotropic Heisenberg granular model, such as the large fluctuations with higher K values. The fluctuations present a difficult problem, as they make finding accurate values of T_c and T'_c more challenging. Different techniques, such as Binder cumulants, could be explored, as they might provide better data. The specific heat and susceptibility are usually well-behaved except for higher values of K and so working with a lower K is desirable.

The Wolff algorithm was again essential for the granular Heisenberg model as it allows the flipping of entire grains at once. As the fluctuations for high K happen around T_c , where the grains start to order, individual grains have spins mostly pointing in a single direction (uniform magnetization). Using only Metropolis, the grains themselves are not

able to flip efficiently and the magnetization might not reach unity at zero temperature. It is possible that a different algorithm might be able to provide a better solution, accounting more explicitly for the grains.

The graphs of Figs. 4.12 and 4.13 represent an effort to obtain what was seen in the Ising model, a regime corresponding to uniformly magnetized grains. However, this case is not as simple as assuming a linear relationship between T_c and J as it was for the Ising case, since the anisotropy changes the relationship, seen in Fig. 4.11. It is found that there is no real linear regime for lower values of J_{eff} , but there does seem to be some agreement between the granular model data and the homogeneous model data for very low $J_{\text{eff}} < 0.1$.

5.3 Future Work

Appendix A describes the code that was used to obtain the data, with possible improvements and expansions that could be used in the future.

This project still has many avenues to explore. While the result already demonstrate the conditions where the uniform grain approximation fails, a more thorough examination of the anisotropic Heisenberg model still needs to be done.

There is still a thorough analysis to be made of the Heisenberg model with anisotropy, as it would be essential to see quantitatively where and for what values of J and K the grains can be assumed uniformly magnetized. Furthermore, certain important effects were not examined. The anisotropic Heisenberg model with an external magnetic field was not studied, though the coercivity is an important property that needs to be studied. This adds yet another dimension to the model, having H_c as a function of J_{eff} , H_{eff} and K_{eff} .

Another effect that remains unstudied is the effect of magnetic dipole-dipole interactions. While nearest-neighbor exchange interactions dominate the energy in the absence of a large external field, this interaction affects every spin of the lattice through an in-

verse power law and can contribute to important effects in thin films. As every spin on the lattice feels the dipole-dipole effect of every other spin, this interaction is very computationally expensive.

A more thorough study of finite size effects and the number of Monte Carlo steps as applied to the granular model could also be done. Ideally, larger L' and L would be used to better represent actual magnetic grains used in recording media. As the study of granular models using Monte Carlo methods represents a different paradigm to simulate magnetic recording media, there are plenty of opportunities for expansion of the project. Usage of the dynamic LLG equation is more prominent as it provides a straightforward to simulate the time dependence of spins, but can be computationally more expensive to achieve equilibrium results. It is possible to create hybrids of the Monte Carlo method with the LLG equation to capture the best of both worlds.

Bibliography

- [1] W. F. Brown *Phys. Rev.*, vol. 130, p. 1677, 1963.
- [2] M. Plumer, J. van Ek, and D. Weller, *The Physics of Ultra-High-Density Magnetic Recording*. Springer-Verlag, 2001.
- [3] G. Grinstein and R. H. Koch *Phys. Rev. Lett.*, vol. 90, p. 207201, 2003.
- [4] O. N. Mryasov, U. Nowak, K. Y. Guslienko, and R. W. Chantrell *Europhys. Lett.*, vol. 69, p. 805, 2005.
- [5] N. Kazantseva, D. Heinzke, U. Nowak, R. W. Chantrell, U. Atxitia, and O. Chubykalo-Fesenko *Phys. Rev. B*, vol. 77, p. 184428, 2008.
- [6] S. Rohart, P. Campiglio, Repain, Y. Nahas, C. Chacon, Y. Girard, J. Lagoute, A. Thiaville, and S. Rousset *Phys. Rev. Lett.*, vol. 104, p. 137202, 2010.
- [7] R. H. Kodama and A. E. Berkowitz *Phys. Rev. B*, vol. 59, p. 6321, 1999.
- [8] X. Z. Cheng, M. B. A. Jalil, and H. K. Lee *IEEE Trans. Magn.*, vol. 43, p. 2899, 2007.
- [9] H. Kachkachi *J. Magn. and Magn. Mat.*, vol. 316, p. 248, 2007.
- [10] J.-G. Zhu, H. Yuan, S. Park, T. Nuhfer, and D. E. Laughlin *IEEE Trans. Magn.*, vol. 45, p. 911, 2009.

- [11] J. Mazo-Zuluga, J. Restrepo, F. Muñoz, and J. Mejía-López *J. Appl. Phys.*, vol. 105, p. 123907, 2009.
- [12] D. A. Garanin and H. Kachkachi *Phys. Rev. B*, vol. 80, p. 014420, 2009.
- [13] M. Plumer, J. van Ek, and W. Cain unpublished.
- [14] R. E. Rottmayer *et al. IEEE Trans. Magn.*, vol. 42, p. 2417, 2006.
- [15] J. W. Toigo *Scientific American*, p. p. 59, May 2000.
- [16] A. Li, D. Wei, and F. Wei *J. Magn. and Magn. Mat.*, vol. 320, p. 3108, 2008.
- [17] A. F. Torabi, J. van Ek, E. Champion, and J. Wang *IEEE Trans. Magn.*, vol. 45, p. 3848, 2009.
- [18] C. Bunce, J. Wu, G. Ju, B. Lu, D. Hinzke, N. Kazantseva, U. Nowak, and R. Chantrell *Phys. Rev. B.*, vol. 81, p. 174428, 2009.
- [19] J. M. Yeomans, *Statistical Mechanics of Phase Transitions*. Oxford, 1992.
- [20] E. Ising *Zeitschrift fr Physik A Hadrons and Nuclei*, vol. 31, pp. 253–258, 1925.
- [21] N. W. Ashcroft and N. D. Mermin, *Solid State Physics*. Thomson Learning, 1976.
- [22] N. Goldenfeld, *Lectures on Phase Transitions and the Renormalization Group*. Advanced Book Program, 1992.
- [23] N. Metropolis and S. Ulam *Journal of the American Statistical Association*, vol. 44, pp. 335–341, 1949.
- [24] M. E. J. Newman and G. T. Barkema, *Monte Carlo Methods in Statistical Physics*. Oxford, 1999.
- [25] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller *J. Chem. Phys.*, vol. 21, pp. 1087–1091, 1953.

- [26] U. Wolff *Phys. Rev. Lett.*, vol. 62, p. 361, 1989.
- [27] L. Onsager *Phys. Rev.*, vol. 65, pp. 117–149, 1944.
- [28] J. A. Plascak, A. M. Ferrenberg, and D. P. Landau *Phys. Rev. E*, p. 066702.
- [29] A. L. Talapov and H. W. J. Blte *J. Phys. A: Math. Gen.*, vol. 29, pp. 5727–5733, 1996.
- [30] X. T. P. Phu, V. T. Ngo, and H. T. Diep *Surface Science*, vol. 603, p. 109, 2009.
- [31] R. K. Pathria, *Statistical Mechanics*. Elsevier, second ed., 1996.
- [32] U. Nowak, R. W. Chantrell, and E. C. Kennedy *Phys. Rev. Lett.*, vol. 84, p. 163, 2000.
- [33] X. Z. Cheng, M. B. A. Jalil, H. K. Lee, and Y. Okabe *Phys. Rev. Lett.*, vol. 96, p. 067208, 2006.
- [34] K. Binder and D. W. Heermann, *Monte Carlo Simulation in Statistical Physics: An introduction*. Springer, fourth ed., 2002.
- [35] M. L. Plumer, J. van Lierop, B. W. Southern, and J. P. Whitehead *J. Phys.: Condens. Matter*, vol. 22, p. 296007, 2010.
- [36] K. Chen, A. M. Ferrenberg, and D. P. Landau *Phys. Rev. B*, vol. 48, p. 3249.
- [37] A. M. Ferrenberg and D. P. Landau *Phys. Rev. Lett.*, vol. 69, p. 3382, 1992.
- [38] W. Selke, A. L. Talapov, and L. N. Shchur *JETP Lett.*, vol. 58, p. 665, 1993.

Appendix A

Code Used in Simulations

The code that was used to obtain these results was written in C++ with a Python front end. An input file is used to set the parameters before the program is run over MCS steps and statistics are taken. Fig. A.1 shows a graphical program that was made with QT that could be used to create the input files for the Heisenberg code.

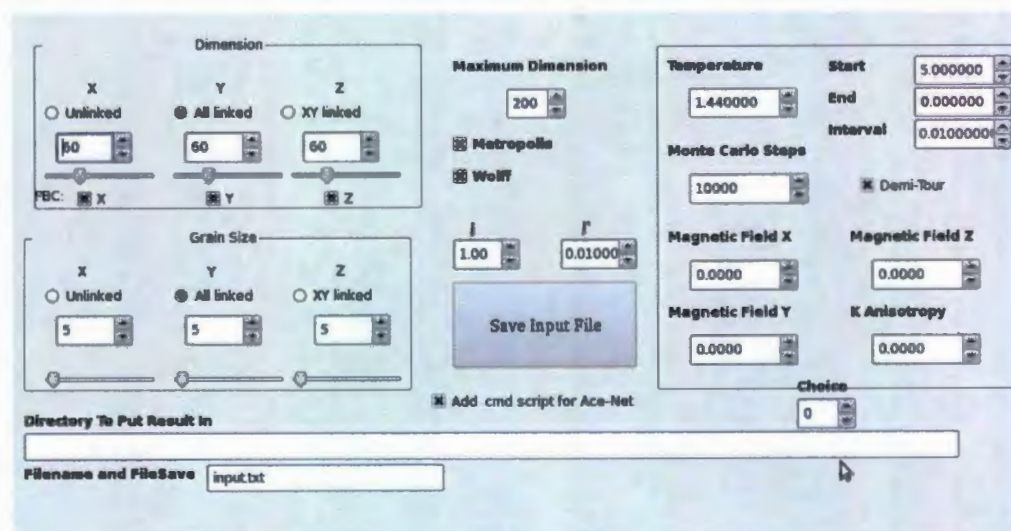


Figure A.1: Input file generator for the Heisenberg model studied with the Monte Carlo method.

Several parameters can be set. While M (and C , χ , etc.) vs T is often used to determine the critical temperatures, M-H loops are done differently. Furthermore, M

vs MCS_t runs are done slightly different. Selection of the specific run is done with the “Choice” parameter. Other important parameters are set as follows:

- The dimension of the lattice LL' is set for the x , y and z dimensions.
- The grain size L' is set for the x , y and z dimensions.
- The periodic boundary conditions (PBC) are added or removed individually. For the multilayer model, it is removed in the z direction.
- The use of the Metropolis and Wolff algorithms are individually set depending on the preference.
- J and J' are set as preferred. Usually, $J = 1$. Setting $J = J' = 1$ and $L' = LL'$ ($L = 1$) will calculate the model for the homogeneous (no grain) case.
- If not doing M vs T runs, the temperature is set to a specific value.
- If not doing M vs MCS_T, the amount of Monte Carlo steps is set.
- If not doing M-H loops, a static magnetic field can be set in any direction (usually z).
- The anisotropy in the z direction is set with K
- The value range to sweep over and the step size is also set with “Start”, “End” and “Interval”. For example, a M vs T run that cools down could be made from $T = 5.0$ to $T = 0.1$ with a step size of 0.1.

This allows to easily save an input file to be used by the main program. As there are many different parameters to try, this program is *trivially parallelized* such that each different iteration of parameters can be run independantly on a different computer core. These programs are run in the high-performance computing environment of the Atlantic Computational Excellence Network (ACEnet), a partnership of multiple Atlantic Canada

universities. Hundreds of jobs can then be run simultaneously, which can each take multiple days. Otherwise, the main program is not parallelized. This is in part due to the use of the Wolff algorithm, which requires information from the entire lattice to properly operate.

Additional programs were made to help with data management. Fig. 2.9 was obtained with a written C++ program that took the spins configurations and created a PNG image from it. Furthermore, numerous scripts were used to help with automating several repetitive tasks such as data formatting, organization and management.

The random number generator used was the Mersenne Twister pseudorandom number generator, which was created with Monte Carlo simulations in mind. The benefits in speed and an extremely long period make it a natural choice. It has been noted in the past that certain bad random number generators can generate errors, particularly with the Wolff algorithm [37, 38].

The Heisenberg code is a fork of the Ising code, which is much simpler and faster, though fundamentally the same. To help with run time speed, several tricks were used. Previously mentioned, the Ising code has a lookup table holding the different possible energy. Consideration of every possible permutation of having $z' = 0$ to 6 nearest neighbors, with 0 to z' of them having a J interaction, gives a total of 210 possible nearest-neighbor energies that can be precalculated before iterating over the main Monte Carlo loop.

The neighbors, with considerations of periodic boundary conditions, can also be computed before going into the main loop and put into another lookup table. This eliminates the need to check if a spin is at the lattice boundary. It also allows the code to be used for other lattice configurations, such as a triangular lattice, with minimal changes, as a spin's neighbors are known before going in the main loop. Similarly, a lookup table can also be done beforehand to know if a spin's neighbor is connected by J or J' , putting the calculation of the determination of a spin's position within a grain outside the main loop. Particular attention must be made to using many lookup

tables, as the program can actually slow down if it becomes *memory bound*, such that it spends most of its time looking in memory for pre-determined values, which can be a slow operation.

Approximate benchmarks of the program are as follows. The Heisenberg code will compute a $10 \times 10 \times 10$ spin cube for $MCS_t = 10000$ with the Metropolis and Wolff algorithm for one single temperature in a little less than one minute of CPU time. This time will depend on many factors, such as the computer the program is run on, but does provide a good guideline, as a similar program with $MCS_t = 20000$ takes approximately twice the time and a cube of $20 \times 20 \times 20$ spins takes approximately eight times as long to run. Having large systems with the Wolff algorithm will noticeably increase the run time, as at lower temperatures, the virtual clusters become pretty big; the algorithm then considers many spins. The Ising code with $120 \times 120 \times 10$ spins, $MCS_t = 25000$, using both the Metropolis and Wolff algorithms for 250 separate temperatures took around 60-90 hours, depending on J and thus depending on how many temperatures the Wolff algorithm was constructing big clusters.



